

***Universidad de las Ciencias Informáticas***

***Facultad 10***



***Sistema de Gestión de materiales audiovisuales para el portal***

***Inter-Nos***

*Trabajo de Diploma para optar por el título de  
Ingeniero en Ciencias Informáticas*

**Autores:** Alexander Martínez Fajardo

Francisco Izquierdo García

**Tutor:** Ing. Sergio Jesús García de la Puente

***Ciudad de la Habana, Cuba.***

***Mayo, 2010***

***Año 52 del Triunfo de la Revolución Cubana***

## Declaración de Autoría

Declaramos ser autores de la presente tesis y reconocemos a la Universidad de las Ciencias Informáticas los derechos patrimoniales de la misma, con carácter exclusivo.

Para que así conste firmo la presente a los \_\_\_\_ días del mes de \_\_\_\_ del año 2010.

Alexander Martínez Fajardo

Francisco Izquierdo García

---

Firma del Autor

---

Firma del Autor

Ing. Sergio Jesús García de la Puente

---

Firma del Tutor

## **Agradecimientos**

*A mi madre que ha sido la persona más importante de mi vida, que ha dado la vida completa para hacer de mi una gran persona, sinónimo de lo que ella considera "su hombrón". La persona que ha estado siempre velando por mi y de quien estoy muy orgulloso por su ejemplo de madre, amiga y esposa.*

*A mi padre por ser para mi un padre que ha sabido darme su cariño y su ejemplo de batalla, sacrificio y tenacidad, alguien que ha sabido conducirme bajo los principios de la honradez y el esfuerzo propio para lograr lo que uno quiere.*

*A mi grandiosa familia y los momentos felices que me han hecho pasar durante todos estos años, porque siempre me brindaron toda su ayuda incondicional, guiándome por el camino correcto. Sin ellos no sería lo que hoy soy.*

*A mis profesores y maestros por su incalculable valor y por ser partícipes de mi preparación profesional.*

*A mis compañeros y amigos por darme la oportunidad de compartir juntos tantos momentos de estudio, tristeza, de molestia, de aburrimiento, de confesiones y felicidad. A todos ellos también llegue mi agradecimiento eterno.*

*A todos aquellas personas que de una forma u otra permitieron lograr este resultado.*

*Alexander Martínez Fajardo*

*A los muchachos del proyecto SIMAYS por sus críticas constructivas y a Yoandy por su ayuda con el OpenOffice.*

*A Javier, por su incalculable ayuda como desarrollador.*

*A Alexander, mi compañero de tesis.*

*A Sergio, nuestro tutor, por sus consejos y guía en el desarrollo de este trabajo.*

*A Alié, Humberto y Camilo y el resto de los muchachos de mi aula por estar al tanto de mi tesis.*

*Francisco Izquierdo García*

## Dedicatoria

*A mis padres que me han sabido dar todo el amor del mundo y que han sido las personas inspiradoras de mis metas, y que ya hoy ven el resultado de su esfuerzo extraordinario.*

*A mi abuela que aunque no estás conmigo físicamente sigues en mi mente y en mi corazón en todo momento, y sé que te hubiese gustado verme en estos momentos. A ti abuela va dedicado el éxito del hombre que ayudaste a crear y por el cual estuvieses muy orgullosa y feliz.*

*A mi familia en general que ha sabido brindarme todo su apoyo y han estado presente en todo los momentos de mi vida.*

*A mis amigos Yoandy, Javier, Camilo y Karel ; de los que nunca ha faltado su mano amiga en las buenas ni en las malas.*

*A mi tutor Sergio que su esfuerzo es parte de mi resultado , y que además de ser mi hermano ha sido mi guía durante toda la investigación.*

*A la Universidad de las Ciencias Informáticas que me dio la oportunidad de formarme como ingeniero y al proyecto UNICORNIOS, que desde sus inicios me acogió y me enseñó a pensar colaborativamente.*

*Alexander Martínez Fajardo*

*A mi papá, por sus consejos de toda la vida, por tratar de inculcarme al menos una pizca de su forma de ser.*

*A mi mamá, por aguantar mis pesadeces y dejarme claro que sobre todo es mi madre.*

*A mi hermanita Gretel, que ya no es tan chiquita, pero sigue siendo mi niñita.*

*A mis tías Eldita, Magnolia y Marisleidis por su preocupación de toda la vida por mis estudios.*

*Al resto de mis familiares, por su preocupación constante por mi.*

*Francisco Izquierdo García*

## Resumen

El presente trabajo de diploma se centra en la necesidad de desarrollar una aplicación web que permita la gestión eficiente de los materiales que se publican en los diferentes espacios del portal Inter-Nos sobre plataforma libre.

Para darle cumplimiento a la presente investigación se realizó un estudio de sistemas de gestión de materiales audiovisuales con el objetivo de identificar los procesos de gestión que automatizan estos sistemas. Se realizó un estudio de las tecnologías, herramientas y lenguajes utilizados en la construcción de aplicaciones web; definiéndose el uso del lenguaje PHP, los frameworks Symfony y JQuery y el Sistema Gestor de Bases de Datos MySQL como soporte al desarrollo de la solución propuesta. Se identificaron y documentaron todas las funcionalidades que el sistema debía cumplir. El proceso de desarrollo de software estuvo guiado por la metodología ágil SXP, indicada para proyectos con requisitos cambiantes y pequeños equipos de desarrollo.

Se logró un sistema web que cumple con los requisitos establecidos por el cliente, haciendo eficiente la gestión de los materiales que se publican en el portal a través de una interfaz de usuario flexible y productiva. Además de la automatización de las tareas de publicación, el sistema brinda un conjunto de servicios web que exponen la información de los materiales publicados, disponibles para el portal Inter-Nos y la comunidad universitaria.

**Palabras claves:** desarrollo ágil, servicios web, sistema de gestión, software libre

# Índice

Introducción.....	1
1 .Capítulo 1. Fundamentación Teórica.....	6
1.1 Conceptos asociados al dominio del problema.....	6
1.2 Sistemas de Gestión de la Información basados en la Web.....	7
1.3 Análisis de sistemas de gestión y difusión de materiales audiovisuales.....	8
1.3.1 Sistemas de gestión de contenidos audiovisuales .....	8
1.3.2 Sistemas de difusión de vídeo y audio en vivo o bajo demanda.....	14
1.3.3 Valoración del estudio realizado.....	16
1.4 Metodología de desarrollo.....	17
1.4.1 Metodologías ágiles.....	18
1.5 Lenguajes, tecnologías y herramientas de desarrollo utilizadas.....	19
1.5.1 Lenguajes.....	19
1.5.2 Tecnologías.....	22
1.5.2.1 Frameworks de Desarrollo.....	23
1.5.2.2 Frameworks para JavaScript.....	24
1.5.2.3 Frameworks para el lenguaje PHP .....	26
1.5.2.4 Sistemas gestores de bases de datos.....	31
1.5.2.5 Servidores de Aplicaciones Web.....	33
1.5.2.6 Graficación de estadísticas.....	35
1.5.2.7 Servicios Web.....	37
1.5.3 Herramientas de soporte al proceso de desarrollo de software.....	40
1.6 Conclusiones parciales.....	42
2 .Capítulo 2 Definición del sistema propuesto.....	43
2.1 Lista de Reserva del Producto (LRP).....	43
2.2 Lista de Historias de Usuario.....	50
2.2.1 Historia de Usuario Gestionar Series.....	50
2.2.2 Historia de Usuario: Gestionar Carteleras.....	54
2.2.3 Historia de Usuario: Gestionar Películas.....	58
2.2.4 Historia de Usuario: Gestionar Calendario de Publicación.....	61

2.2.5 Historia de Usuario: Gestionar Programas Grabados.....	64
2.2.6 Historia de Usuario: Gestionar Estaciones de Radio y TV en vivo.....	69
2.2.7 Historia de Usuario: Gestionar Teleclases.....	72
2.2.8 Historia de Usuario: Gestionar Protocolos de Transmisión.....	78
2.2.9 Historia de Usuario: Gestionar Usuarios del Sistema.....	80
2.2.10 Historia de Usuario: Servicios Web.....	83
2.3 Plan de Releases.....	84
2.4 Conclusiones parciales.....	85
3 .Capítulo 3 Desarrollo del sistema propuesto.....	86
3.1 Patrones de diseño y arquitectura del framework Symfony.....	86
3.1.1 Patrones de diseño implementados por Symfony.....	86
3.1.2 Implementación del patrón Modelo Vista Controlador de Symfony.....	87
3.1.3 Estructura del framework Symfony.....	88
3.2 Seguridad en las aplicaciones desarrolladas con Symfony.....	89
3.3 Arquitectura de la solución propuesta.....	92
3.3.1 Diagrama de componentes.....	92
3.3.2 Esquema de la base de datos utilizada en la solución propuesta.....	98
3.4 Descripción de las interfaces de comunicación externas (Servicios Web).....	99
3.4.1 Complemento ckWebServicePlugin.....	99
3.4.2 Tipos de datos complejos usados.....	100
3.5 Estándar de codificación utilizado.....	106
3.6 Conclusiones parciales.....	106
4 .Capítulo 4 Validación de la solución propuesta.....	107
4.1 Pruebas funcionales.....	107
4.1.1 Pruebas funcionales realizadas al sistema.....	107
4.2 Impacto y aporte de la solución propuesta.....	112
4.3 Conclusiones parciales.....	113
Conclusiones.....	114
Recomendaciones.....	115
Referencias Bibliográficas.....	116

Bibliografía.....	119
Anexos.....	122
Anexo I Estándar de codificación utilizado.....	122
Anexo II: Diagrama Entidad-Relación. Base de datos internos.....	126

## Introducción

Las tecnologías de la comunicación han revolucionado la forma de difundir el conocimiento en el mundo. Es incuestionable el cambio que han producido y de tal magnitud que ya forman parte de la cultura tecnológica del hombre, ampliando sus capacidades físicas y mentales y las posibilidades de desarrollo social. Es la manifestación audiovisual una de las formas en que el hombre puede comunicarse e interrelacionarse; y a pesar de la existencia de la televisión convencional no es posible dejar de mencionar que “Internet” ha constituido una forma de transmisión de contenidos audiovisuales de incalculable valor.

La inmensa mayoría de los medios de comunicación han sido trasladados de formato duro a formato digital, disponibles en la red a través de la Web. Frente al resto de los medios convencionales, la prensa es la que mejor ha sabido adaptarse a la “Red de Redes”. Quizás la novedad del medio, o el desconocimiento de sus posibilidades, del mercado y de las capacidades, ha retardado el salto tecnológico.

No obstante, las empresas y los medios de comunicación fueron percatándose de las potencialidades que brindaba, ya sea por su facilidad de búsqueda, interactividad, escalabilidad y potencial de difusión de contenidos multimedia. Es así que han encontrado en Internet su principal aliado, desempeñando su papel como vehículo de la comunicación en el mundo. De tal forma se abrieron las puertas a la diseminación e intercambio de conocimientos en sus diferentes manifestaciones audiovisuales [1].

Es de esta manera que el hombre ha ido marcando pautas en el largo camino de la comunicación audiovisual. Programas como los reproductores y servidores de media streaming, término que “se aplica a una serie de productos y técnicas cuyo objetivo es la difusión de contenidos multimedia tales como audio y vídeo” [2]; han condicionado el aprovechamiento del ancho de banda y el tráfico de la red específico en cada espacio. Ya es posible recibir imagen y audio en tiempo real, y transmitir por Internet o Intranet, un concierto de música, un programa de noticias de televisión o una serie creada específicamente para Internet. De modo que es de vital importancia documentar, organizar, clasificar, y elaborar la ficha descriptiva de cada material que se visualizará en la red, ya sea bajo demanda o en vivo; de esta manera sería más fácil realizar búsquedas y repeticiones de los mismos y no estarán disgregados como un simple almacén de audiovisuales.

Actualmente la Universidad de las Ciencias Informáticas (UCI) cuenta con un portal web para la

publicación de materiales audiovisuales de corte educativo, recreativo y docente llamado “Inter-Nos”. La información publicada en el portal es gestionada a través de una aplicación con una interfaz poco flexible y productiva, a causa de su estructura ineficiente y poca navegabilidad. Asimismo, no abarca todas las necesidades de gestión de la información publicada en Inter-Nos, que en ocasiones hace necesario el conocimiento por parte de los administradores sobre temas de programación web y bases de datos para realizar cambios en la información mostrada. No existe un calendario en el sistema que permita automatizar las tareas de publicación.

Todas estas razones provocan retrasos en la actualización de algunos de los espacios más solicitados e insatisfacción en la comunidad universitaria. El actual sistema de gestión está soportado sobre plataforma privativa (ASP, IIS, SQL-Server 2000) y la Universidad como parte de su estrategia productiva está enfrascada en la migración del mismo a plataformas libres. De acuerdo a la situación descrita anteriormente se identifica como **problema científico**, ¿cómo hacer eficiente la gestión de los materiales audiovisuales de los diferentes espacios del portal Inter-Nos?

Estableciéndose como **objeto de estudio** el proceso de gestión de contenidos audiovisuales y como **campo de acción** la gestión de la información audiovisual del portal Inter-Nos en la Universidad de las Ciencias Informáticas, presentándose como **objetivo general** desarrollar un sistema de gestión de contenidos audiovisuales sobre entornos con software libre para el portal Inter-Nos. Definiendo como **idea a defender**, la siguiente: con la elaboración de una aplicación de gestión de contenidos audiovisuales desarrollada con tecnologías libres y que brinde una interfaz flexible y acorde a los requisitos del cliente, se agilizarán los procesos de gestión de los diferentes espacios del nuevo portal Inter-Nos.

Para lograr el objetivo planteado anteriormente se determinaron varios **objetivos específicos**.

- Caracterizar las tendencias actuales sobre las soluciones de gestión de contenidos audiovisuales.
- Caracterizar las tecnologías involucradas en la creación de los servicios web que brindará la solución propuesta.
- Definir las herramientas libres que soportarán el desarrollo y futuro despliegue de la solución propuesta.

- Documentar las funcionalidades del sistema definidas por el cliente.
- Implementar las funcionalidades definidas por el cliente.
- Validar la solución desarrollada.

Para darle cumplimiento a los objetivos trazados se han propuesto las siguientes **tareas de investigación**:

- Revisión y análisis de la bibliografía existente sobre algunas de las soluciones de gestión de materiales audiovisuales.
- Realización de un estudio de las tecnologías referentes a la creación de servicios web.
- Estudio de las herramientas libres a utilizar en el desarrollo de la solución propuesta.
- Estudio de la metodología de desarrollo empleada.
- Descripción de la arquitectura del sistema y patrones de diseño empleados.
- Diseño y ejecución de pruebas funcionales a la aplicación para asegurar la calidad del producto.

La realización de las tareas de investigación son concretadas sobre la base del uso de **métodos de investigación científica**. Estos permitieron conocer la esencia del problema abordado y las relaciones de los procesos de gestión de materiales audiovisuales que se llevan a cabo para brindar el servicio de publicación de información en Inter-Nos. Se dividen en empíricos y teóricos. Los métodos teóricos permiten estudiar las características del objeto de investigación que no son observables y crean las condiciones para ir mas allá de las características fenomenológicas y superficiales de la realidad.

Los métodos teóricos empleados en la presente investigación son el analítico-sintético y el inductivo-deductivo. El método analítico-sintético se enfoca en determinar la esencia de los fenómenos, los rasgos que los caracterizan y los distinguen. Este método fue utilizado en la determinación de las características comunes y específicas de los sistemas que gestionan materiales audiovisuales para definir los procesos de gestión que automatizan. El método inductivo-deductivo permite llegar a proposiciones generales a partir de hechos aislados que confirman la teoría o a partir de estas teorías arribar a conclusiones sobre casos particulares que se verifican en la práctica. Este método se utiliza

con el fin de inferir a partir de las aplicaciones dedicadas a la publicación de contenidos audiovisuales la manera en que se gestionan estos contenidos.

Se hace uso además, de la entrevista como técnica de recolección de datos. “La entrevista es una conversación planificada entre el investigador y el entrevistado para obtener información. Su uso constituye un medio para el conocimiento cualitativo de los fenómenos o sobre características personales del entrevistado y puede influir en determinados aspectos de la conducta humana por lo que es importante una buena comunicación” [3].

Se realizaron entrevistas no estructuradas al cliente, especialista en el tema de publicación de contenidos audiovisuales, con el objetivo de comprender la situación de la aplicación de gestión actual y profundizar en los detalles de las nuevas funcionalidades de la aplicación.

El documento está estructurado de la siguiente manera:

- **Capítulo 1: Fundamentación Teórica:** Estudio del estado actual del arte sobre sistemas de gestión de contenidos audiovisuales, sus principales funcionalidades y servicios a las comunidades. Análisis del estudio realizado y conclusiones sobre el sistema que se desarrollará. Argumentación de la Metodología y Tecnología de desarrollo empleada para el desarrollo del sistema propuesto.
- **Capítulo 2: Definición del sistema propuesto:** Identificación de las funcionalidades del sistema, así como de los componentes a desarrollar y su planificación.
- **Capítulo 3: Desarrollo del sistema propuesto:** Presenta la descripción de la arquitectura del sistema, empleo de patrones arquitectónicos y la estrategia de integración de los componentes. Se describe la estandarización del código. Descripción de las interfaces de comunicación con el sistema propuesto (servicios web).
- **Capítulo 4: Validación de la solución propuesta:** Descripción de los casos de pruebas. Análisis del impacto de la solución propuesta.

La presente investigación aportará servicios web disponibles para la comunidad universitaria, que pueden ser consumidos desde otras aplicaciones que se desarrollen posteriormente. Unido a esto, mejorará los niveles de eficiencia de los procesos de gestión de los materiales audiovisuales que se publican en el portal Inter-Nos.

## **Capítulo 1. Fundamentación Teórica**

En el presente capítulo serán analizados los fundamentos de la gestión de contenidos audiovisuales; así como algunas soluciones de sistemas de gestión de contenidos audiovisuales existentes en el mundo. Se muestran además las diferentes herramientas, tecnologías y metodología utilizadas en la implementación del sistema.

### **1.1 Conceptos asociados al dominio del problema**

#### **Gestión de la Información**

En la era de la información, donde las tecnologías han ocupado casi todas las esferas del desarrollo del hombre, el conocimiento es la fuente de riquezas y el factor esencial del progreso. Las tecnologías de la información y las comunicaciones vienen a componer el medio en el que se difunde y gestiona la información, elemento esencial en el desarrollo; y su clasificación, control y acceso son de vital importancia no solo para las empresas, sino para las economías de los países a nivel mundial.

Se entiende como gestión de la información al “medio por el cual una organización incrementa la eficiencia con la cual planea, organiza, usa, controla, almacena, disemina y dispone de su información, y a través de la cual asegura que el valor de esta información es identificado y explotado al máximo posible. El término es referido a veces a la gestión de la información empresarial, gestión de los recursos de la información, o la inteligencia de negocio, especialmente en conexión con programas de computación relevantes [4]. El proceso de gestión de la información va aparejado al mejoramiento de los productos y servicios de una organización, y en concordancia con la actividad fundamental a la que las empresas dedican sus esfuerzos.

La gestión de la información reviste una enorme importancia, de ahí que en general se toma como una buena práctica en todas las esferas del desarrollo, contribuyendo a la claridad y velocidad en el trabajo, búsqueda rápida de información y ganancia de tiempo. De igual forma, las empresas ya no pueden manejar demasiada información manualmente, que unido a la inadecuada gestión de los procesos empresariales ocasiona un mal funcionamiento de las entidades. El exceso de información es un factor clave que define la necesidad de su gestión, lo cual enfatizado con la Internet, que permite a cualquier persona publicar o introducir contenido en la Web, se mezcla la información redundante y de baja calidad con información veraz y sólida.

## **Aplicaciones web**

La revolución de la Era Digital y las comunicaciones, estuvo marcada fuertemente por la interconexión de miles de ordenadores en el mundo, a lo que comúnmente se le denomina Internet, que impulsó el surgimiento de la World Wide Web (WWW). Aparecen las páginas web que no son más que un conjunto de documentos enlazados a través de hipertextos, y surgen las llamadas aplicaciones web.

Una aplicación web, es un conjunto de páginas web con contenido sin determinar, parcialmente o en su totalidad y que se genera en el momento en que el usuario realiza la solicitud a un servidor web, en el que se hospeda dicha aplicación. Las páginas que se generan como respuesta del servidor web a la solicitud del usuario son páginas dinámicas, dado que su contenido varía de una petición a otra. La característica que define a una aplicación web, es el uso de las componentes que definen la arquitectura de la Web: HTTP y URI [5]. De acuerdo a la finalidad de las aplicaciones web con respecto a la información que disponen, estas se clasifican en aplicaciones front-end y back-end; las primeras son aquellas con las que interactúa el usuario final a través de interfaces amigables, y las segundas constituyen el área de administración, donde se ingresa, edita y organiza el contenido que será visualizado en el front-end.

### **1.2 Sistemas de Gestión de la Información basados en la Web.**

Una de las áreas en las que más actúa la informática, lo constituye la gestión de la información a través de sistemas que automatizan estos procesos. Aprovechando las potencialidades que brindan las aplicaciones web, por su portabilidad, escalabilidad y accesibilidad al contenido, han tomado auge los sistemas de gestión de información online. Los mismos permiten trabajar en un entorno unificado y centralizado, con acceso seguro e inmediato a los datos y las tareas de interés para cada organización.

Al tener la información en un servidor web, se puede acceder a ella desde cualquier dispositivo con acceso a Internet. Por lo general estos sistemas están dedicados a determinada necesidad de gestión, ya sea gestión de la contabilidad, ventas, gestión académica, contenidos audiovisuales, entre otros. Esta especificidad provoca que no sean reutilizables la mayoría de las veces en otros entornos y es difícil adaptarlos a otros procesos de gestión de información.

## **Sistemas de gestión de contenido (CMS)**

Precisamente, existen tecnologías o herramientas que permiten una gestión de los contenidos que son presentados al usuario en un sitio o aplicación web. La peculiar característica de estas tecnologías radica en que son aplicables prácticamente a cualquier necesidad de gestión que afronte una empresa o institución. A estas aplicaciones web se les denomina Sistemas de Gestión de Contenidos (del inglés, Content Management Systems).

“Un CMS es una herramienta que permite a un editor crear, clasificar y publicar cualquier tipo de información en una página web. Generalmente los CMS trabajan contra una base de datos, de modo que el editor simplemente actualiza una base de datos, incluyendo nueva información o editando la existente” [6]. De acuerdo a la utilización de los mismos, se clasifican en empresariales, foros, blogs, wikis, e-learning, entre otros. Otra clasificación es de acuerdo a la propiedad del código, ya sea código abierto o cerrado.

### **Ventajas del uso de los CMS**

Los Sistemas Gestores de Contenidos son un ejemplo de aplicaciones que integran el front-end y el back-end en un paquete único, y posibilitan la creación y administración de sitios web dinámicos de manera rápida. Proveen herramientas para gestionar el ciclo de vida de los contenidos, ejemplo herramientas de edición WYSIWYG (del inglés, What you See Is What You Get), en el que el editor ve el resultado final mientras escribe. Posee una estructura modular que permite incluir nuevas funcionalidades. Otra ventaja radica en la fuerte política de seguridad basada en roles para la gestión y publicación de la información [7].

## **1.3 Análisis de sistemas de gestión y difusión de materiales audiovisuales**

### **1.3.1 Sistemas de gestión de contenidos audiovisuales**

Los sistemas de gestión de información audiovisual son aquellos que se encargan de manera general de los procesos de ingesta, conversión, edición, catalogación o documentación, clipping y publicación de audio o vídeo, para ser transmitido en vivo o bajo demanda (contenidos solicitados por el usuario). El flujo de trabajo consta de varios pasos:

- **Ingesta:** proceso mediante el cual se inserta el contenido digital de forma automatizada sin necesidad de supervisión mediante directorios de escucha o “watchfolder”.

- **Conversión:** Conversión automática de formatos en alta calidad a calidades de visionado web.
- **Documentar:** Documentación del material de vídeo o audio a través de metadatos que almacenan los datos técnicos o solo información.
- **Búsqueda:** Parámetros de búsqueda configurables por el usuario para satisfacer una necesidad de información.
- **Clipping:** Fragmentación del vídeo en segmentos (clips).
- **Publicación Web:** proceso mediante el cual se pone a disposición del usuario el material audiovisual, capturado, editado y catalogado [8].

El estudio realizado aporta elementos que caracterizan los sistemas que automatizan la gestión de materiales audiovisuales. También está enfocado al análisis de algunas aplicaciones front-end que brindan servicio de streaming y descarga de audiovisuales a través de Internet; identificando la manera en que está organizada la información que presentan, con el fin de tener una idea de cómo se gestiona dicha información.

#### ➤ **Magnolia On Air**

Es una solución nueva y dedicada para los medios de difusión y grandes organizaciones que manejan su propia emisión de contenido. Magnolia-on-Air provee un ambiente ininterrumpido de transferencia de contenido entre sistemas profesionales de difusión y una presencia web interactiva. Basado en Magnolia CMS, el sistema gestor de contenido de código abierto, del cual Magnolia On Air adopta muchas de las funcionalidades y características expuestas.

#### **Características:**

- El contenido de difusión puede ser enriquecido con material adicional de la comunidad, de una editorial o de los publicadores. Esto no lo soportan en el medio de difusión tradicional.
- Altamente extensible debido a su licencia de Código Abierto.
- La interfaz de administración de Magnolia usa las tecnologías AJAX para proveer gestión de contenido intuitivo y una experiencia de usuario agradable.
- Se ejecuta en todos los sistemas operativos comunes, debido a que está construido sobre plataforma Java y es funcional en cualquier navegador.

- La interfaz de usuario de Magnolia On Air está disponible en más de 15 idiomas, entre los que se incluyen inglés, ruso y español.
- Posee una gran escalabilidad, la capacidad de almacenaje puede ser aumentada poniendo más servidores dedicados y soporta el balanceo de carga.
- Control de acceso basado en roles.

#### **Estándares:**

- Apoya el estándar JSR-170, el cual separa las responsabilidades de almacenamiento de contenido de la de gestión de contenidos, y se adhiere a estándares abiertos como XHTML, Java, ECMA-Script, JSP, J2EE, etc.
- Accesibilidad Conforme a W3C: las plantillas de Magnolia siguen las guías estipuladas por el W3C con respecto a la accesibilidad de contenido (WCAG2m) así como las guías alemanas del BITV (Barrierefreie Informations Technik-Verordnung).
- Soporta multitud de formatos profesionales de difusión de entrada y salida (MXF, MPEG2, MPEG4, H.264, AAC, MP3, WMV, WMA, IDE, Flash, Quicktime).

#### **Funcionalidades:**

- Captura, edición y publicación de contenido multimedia a través de una interfaz de usuario basada en la web.
- Contenido editorial adicional y contenido generado por el usuario, fomenta la retención de usuarios y la participación (retroalimentación).
- Interfaz de usuario fácil de usar, basada en la web que permite readaptar y reenvasar contenidos de difusión de manera rápida para la web y editar archivos de vídeo y audio antes de publicarlos.
- Comunicación interactiva: El contenido puede ser enriquecido con otro contenido editorial, con contenido creado por la comunidad o con material de los archivos de la editorial.
- Gestión de contenido de múltiples formatos y múltiples canales.
- Publicación basada en calendarios.
- Permite que se trabaje en flujos de trabajo a dos niveles: un primer nivel en el que se realiza el flujo de edición y un segundo nivel donde una segunda persona pueda verificar el contenido antes

de que este sea activado. Magnolia CMS en su edición Enterprise permite opciones de flujos de trabajo adicionales.

### **Requisitos de Software:**

Está disponible como un paquete autónomo de Apache Tomcat (incluye servidor de aplicaciones, Magnolia CMS, repositorio de contenido), o como un archivo .war desplegable drop-in J2EE. Ningún otro software o bases de datos son requeridos [9].

#### ➤ **Videoma Tube**

Es una solución para la publicación de contenido multimedia en la web 2.0. Basado en el sistema web Videoma Archivo para la gestión de videotecas y permite compartir vídeos con una comunidad virtual. Realiza la ingesta automática de vídeo, catalogación, clipping y las llamadas a la base de datos para la publicación web mediante el Módulo API de Videoma. Automatiza los procesos de gestión de materiales audiovisuales descritos anteriormente.

### **Configuración de Hardware:**

- Procesador: dual Core o superior.
- Memoria RAM: 2 GB o superior.
- Almacenamiento: 1 TB. Capacidad aproximada para 500 hrs a 4 MB/sg.

### **Estándares:**

- Al estar basado en Videoma Archivo incorpora diferentes estándares y tecnologías: Servicios Web bajo estándar SOAP.
- Interoperabilidad con lenguaje escogido siempre que disponga de librerías de conectividad de servicios SOAP.
- Para representar los datos y el intercambio de información y contenido, emplea estándares como MPEG-7, el cual es utilizado para la anotación, descripción y categorización de los materiales audiovisuales.
- El W3C XML 9 (Extensible Mark-up Language), lenguaje de marcado que utiliza para la creación de esquemas de descripción y descriptores.
- Utiliza el lenguaje de marcado basado en XML de la W3C SMIL, Integración multimedia

sincronizada de lenguaje (del inglés, Synchronized Multimedia Integration Language), específicamente para la manipulación y posicionamiento de medios audiovisuales en pantalla, así como para la presentación de medios conforme las preferencias del usuario.

- En el proceso de codificación de vídeos utiliza los estándares Windows Media, RealVideo y QuickTime; también estándares de la ISO como MPEG-1, MPEG-2 y MPEG-4.

#### **Características:**

- Sistema Operativo: Microsoft Windows.
- Acceso vía LAN/INTERNET usando un navegador web.
- Gestión de permisos de usuarios.
- Configuración de borrados automatizados.
- Reproductores con controles avanzados.
- Generación de listados de las búsquedas realizadas en formato pdf o txt.

#### ➤ **OntoMedia**

Es un sistema web creado para la gestión multimedia, basado en la tecnología de la web semántica. El principal objetivo de OntoMedia es la gestión de grandes volúmenes de contenidos multimedia usando técnicas de integración semántica para metadatos, aplicando la ontología estado del arte manejada en la tecnología de la web semántica al dominio multimedia.

#### **Características:**

El usuario puede organizar colecciones de diferentes tipos de materiales a través de una intuitiva interfaz de usuario gráfica, que incluye el indexado fácil de metadatos y capacidades de búsqueda. Funciona sobre la plataforma Java y es de código abierto.

#### **Funcionalidades:**

- Escaneo de archivos y extracción de metadatos: provee una vista detallada de metadatos extraídos al escanear los archivos seleccionados por el usuario.
- Organización de archivos: organización gráfica realizada usando un navegador de categorías. En este navegador de categorías se pueden crear categorías por el usuario y luego asignar a cada material multimedia su categoría.

- Búsqueda de contenidos multimedia: la búsqueda es soportada por un rápido motor de búsqueda basado en concepto, el cual indexa todos los metadatos extraídos desde archivos de audio, vídeo u otros.
- Visualizar archivos: incluye algunos componentes internos para reproducir materiales de vídeo y sonido en la aplicación, así como ver imágenes en vista de miniaturas o en una presentación de diapositivas.

### **Estándares:**

Soporta múltiples contenedores de audio y vídeo estándares: Video RIFF (AVI).

- Música: Mp3, WAV.
- Documentos: documentos de la suite ofimática Microsoft Office.
- Imágenes: GIF, JPG, EXIF [10].

### ➤ **Distributed MultiMedia Manager**

Es un sistema distribuido de gestión multimedia que soporta la captura, análisis, recuperación, creación y presentación en único entorno de trabajo. La arquitectura distribuida cliente-servidor es adoptada en este sistema de escritorio, aunque las aplicaciones cliente y servidor fueron desarrolladas usando Visual C++ y Java respectivamente.

### **Funcionalidades:**

- Captura de vídeo y audio en vivo.
- Extracción de características de imagen y análisis de vídeo (shot detection).
- Recuperación de imagen basado en contenido.
- Desplazamiento dentro de un material de vídeo basado en los fotogramas claves dentro del mismo.
- Diseño de la presentación multimedia a través de una Red de Transición de Multimedia Aumentada.
- El Java Media Framework provee la mayoría de las funcionalidades de renderizado y reproducción multimedia en OntoMedia. Usa además codificadores y decodificadores como: QuickTime for Java, 3vix Decoders y JLayer MP3 Framework para reproducir archivos específicos.

- Generación de la presentación online usando SMIL (Synchronized Multimedia Integration Language).
- Distribución de datos usando las redes de computadoras destinadas [11].

### 1.3.2 Sistemas de difusión de vídeo y audio en vivo o bajo demanda

#### ➤ **MovieFlix.com**<sup>1</sup>

Es uno de los líderes proveedores de servicio de streaming y descarga de películas en la web. Fundada en 1998, esta compañía privada ha colaborado con empresas líderes en la industria como Microsoft, RealNetworks, Screaming Media, Yahoo y otras. Dentro de sus proyectos triunfales está el sitio <http://www.movieflix.com>, dedicado a este servicio.

#### **Características:**

- El portal organiza las películas por categorías o géneros.
- Dentro de cada género cinematográfico, muestra un listado de películas ordenadas alfabéticamente por el nombre.
- De cada película muestra además del nombre, el título, sinopsis, año de realización, duración y precio.
- Posibilita la descarga en concordancia con el nivel de conectividad que tenga el usuario.
- Incorpora un buscador interno para facilitar la búsqueda de materiales.
- Muestra estadísticas sobre los materiales más descargados.
- El sitio está hospedado en un servidor Web Apache, versión 3.0.63 y lenguaje de programación PHP versión 4.4.8.

#### ➤ **Online Radioz**<sup>2</sup>

Es un sitio dedicado a la publicación de emisoras de radio, con transmisión en vivo. En la portada del mismo muestra un logo representativo de cada emisora publicada, y al hacer clic sobre una emisora determinada comienza la transmisión de la misma.

#### **Características:**

---

1 <http://www.movieflix.com>

2 <http://nl.onlineradioz.com>

- El portal organiza las emisoras por géneros y por las más populares.
- Ofrece la posibilidad de hacer una búsqueda de emisoras.
- Permite al visitante crear una cuenta de usuario, sugerir nuevas estaciones de radio y organizar sus favoritas.
- Está desarrollado con Symfony, aunque es de código cerrado [12].
- Alojado en un servidor web Apache2 y PHP versión 5.3 como lenguaje de programación del lado del servidor.

➤ **GET Movie<sup>3</sup>**

Es un sitio dedicado a la publicación de carteleras de películas de estreno a nivel mundial.

**Características:**

- Muestra un catálogo de películas organizadas por géneros cinematográficos.
- De cada película muestra un póster y al hacer clic sobre la imagen se visualizan los detalles de la película.
- De cada película muestra además el nombre, género, actor(es), director(es) y sinopsis.
- El sitio da la posibilidad de descargar la película desde URLs externas.
- Permite a los usuarios registrados escribir comentarios sobre determinada película.
- La aplicación está alojada en un servidor web Apache versión 2.2.3 y desarrollada con PHP versión 5.3, sobre sistema operativo CentOS [13].

➤ **Elmulticine.com<sup>4</sup>**

Es una revista digital sobre cine publicada en Internet y distribuida mediante diversos medios telemáticos. El acceso a toda la información publicada es totalmente gratuito y está abierto a visitantes de todo el mundo sin restricciones regionales o direccionamientos específicos de ningún tipo.

---

<sup>3</sup> <http://www.get.vn>

<sup>4</sup> <http://www.elmulticine.com>

### **Características:**

- Muestra en uno de sus espacios una cartelera de estrenos de películas para fechas próximas.
- De cada película publica nombre, director(es), sinopsis, una foto identificativa y un vínculo para visualizar un tráiler de manera online, en caso de estar disponible mediante un reproductor embebido en la propia web.
- Posibilita por cada película publicar una encuesta para recoger criterios de los internautas.
- Está desarrollado con PHP versión 5.1.6, corre sobre un servidor Apache versión 2.2.3 en un sistema operativo CentOS [14].

#### ➤ **Sensacine.com**<sup>5</sup>

Este portal está dedicado a publicar carteleras de estrenos de películas que saldrán próximamente en cines españoles.

### **Características:**

- De cada película muestra una ficha descriptiva, la crítica de la prensa y de los espectadores, así como noticias sobre la misma.
- Brinda la posibilidad de visualizar un tráiler, además de publicar un top list con los estrenos de la semana.
- Tiene una sección de series de televisión que muestra la salida al aire (hora y día) de los últimos capítulos de las series más populares y una imagen identificativa de éstas.
- El sistema está hospedado en un servidor web IIS (Internet Information Server) con tecnología ASP.NET sobre plataforma Windows [15].

### **1.3.3 Valoración del estudio realizado**

Luego de estudiar algunos de los sistemas de gestión de contenidos multimedia existentes y algunos portales dedicados a la publicación de materiales audiovisuales y en concordancia con las funcionalidades que proveen; se puede concluir que no están acorde con las necesidades de gestión de la información que se publica en el portal Inter-Nos de la Universidad de las Ciencias Informáticas.

Algunos de estos sistemas están basados en plataforma privativa (Windows) y otros aunque están

---

5 <http://www.sensacine.com>

disponibles en versiones empresariales y comunitarias, las empresas que las desarrollan dedican sus esfuerzos a las versiones empresariales, por constituir un producto negociable. El CMS Magnolia On Air en su versión comunitaria pudiera cumplir con los requerimientos de gestión del portal Inter-Nos; pero tiene la agravante de que al ser un sistema implementado sobre plataforma J2EE, que captura y edita el archivo de vídeo a publicar a través de una interfaz basada en la web, presupone la necesidad de recursos hardware con mayores prestaciones y una infraestructura de red con más ancho de banda del que dispone la universidad. Asimismo, posee funcionalidades limitadas y restricciones de uso, y el código fuente del mismo está publicado en el repositorio de software SourceForge<sup>6</sup>, sitio bloqueado actualmente para Cuba.

En correspondencia con el estudio realizado sobre los portales que brindan servicio de streaming, se puede concluir que el proceso de catalogación de la información es similar. De estos materiales se almacena generalmente una sinopsis, título, género cinematográfico, actores, directores, duración de la película y una imagen identificativa de la misma; característica que se tomará en cuenta para el desarrollo de la solución. De los procesos de gestión analizados se aplicarán los correspondientes a la catalogación y publicación de materiales audiovisuales.

#### **1.4 Metodología de desarrollo**

La selección de una metodología de desarrollo de software así como las herramientas que soportarán al mismo es una tarea necesaria e importante; la calidad del proceso y del producto final estarán condicionadas en gran medida por estos dos aspectos (herramientas y metodología).

##### **¿ Qué es una metodología de desarrollo de software ?**

“Se entiende por metodología de desarrollo una colección de documentación formal referente a los procesos, las políticas y los procedimientos que intervienen en el desarrollo del software.

La finalidad de una metodología de desarrollo es garantizar la eficacia (p.ej. cumplir los requisitos iniciales) y la eficiencia (p.ej. minimizar las pérdidas de tiempo) en el proceso de generación de software” [16].

Existen muchas metodologías de desarrollo de software, algunas de ellas llamadas “tradicionales”, las cuales ayudan a los profesionales a documentar y realizar las tareas de desarrollo, pero estas tienen como defecto común la carga de burocracia, reiteraciones y ambigüedades. Son rígidas y dirigidas

---

6 <http://sourceforge.net>

por la documentación que se genera en cada una de las actividades desarrolladas. Estas metodologías están prácticamente obsoletas; otras más actuales han sido adoptadas para guiar el proceso de desarrollo, denominadas “metodologías modernas”, dentro de las cuales figuran las “metodologías ágiles”.

### **1.4.1 Metodologías ágiles**

Surgen como una alternativa a las metodologías tradicionales. Su filosofía se centra en el individuo, la colaboración con el cliente, que éste sea partícipe del proceso de desarrollo del software, y el desarrollo incremental del producto software en sí con iteraciones cortas. Se ajustan a proyectos de corta duración y cuya planificación no debe ser estricta, de modo que responda a los cambios (requisitos, tecnologías, etc) que pueden ocurrir a lo largo del proyecto [17].

#### **➤ SXP**

Metodología ágil desarrollada en la Universidad de las Ciencias Informáticas, como un híbrido de SCRUM y XP. Ofrece una estrategia tecnológica, a partir de la introducción de procedimientos ágiles que permitan actualizar los procesos de software para el mejoramiento de la actividad productiva fomentando el desarrollo de la creatividad, aumentando el nivel de preocupación y responsabilidad de los miembros del equipo, ayudando al líder del proyecto a tener un mejor control del mismo, y cuya particularidad es tener como parte del equipo, al usuario final, pues es uno de los requisitos para llegar al éxito del proyecto. Con el uso de SCRUM para la gestión del trabajo, SXP logra resultados satisfactorios y toma de XP las mejores prácticas que guían el desarrollo del software, como la refactorización y pruebas continuas.

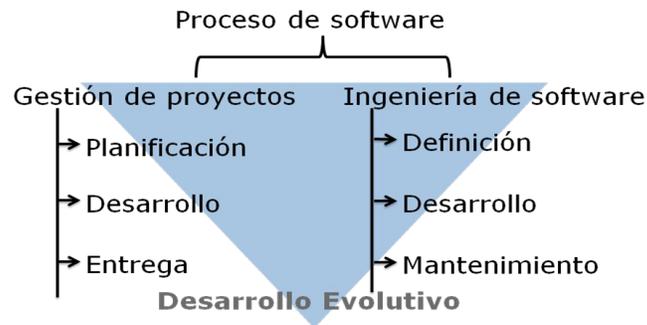
#### **Un poco de Historia**

SXP surge producto a la necesidad de establecer una guía ágil para los procesos de desarrollo de software libre en la Facultad 10 de la Universidad de las Ciencias Informáticas. Inicialmente hacia el año 2007 la Ing. Malay Rodríguez ya había realizado una investigación en la Facultad 7 en este sentido, que arrojó como resultado la propuesta MA-MRV-R1. Posteriormente ponen en práctica esta propuesta seis sistemas escogidos del grupo UNICORNIOS de la Facultad 10, de la Universidad de las Ciencias informáticas.

A partir de entonces se hace un estudio de esta propuesta y se logra la propuesta de la metodología MA-GMPR-UR2 hacia el año 2008. En el grupo UNICORNIOS se comienza a usar esta metodología

en este mismo año, por lo que cuenta con aproximadamente 2 años de implantación; de ahí que el proceso de desarrollo de la solución brindada con la presente investigación esté guiada por esta metodología.

SXP consta de 3 fases, las que se muestran en el gráfico de la **Figura 1.1**.



*Figura 1.1: Esquema de la metodología SXP*

- **Fase de Planificación-Definición:** su propósito es establecer la visión, fijar expectativas y asegurar financiamiento.
- **Fase de Desarrollo:** el propósito es implementar un sistema listo para entrega en una serie de iteraciones de 60 días. (El tiempo puede decrementarse a la medida que se está refinando el producto).
- **Fase de Entrega-Mantenimiento:** se entrega el producto y se le da el soporte necesario.

SXP es ideal para proyectos de corta duración con requisitos cambiantes o no bien definidos, donde prevalezca la retroalimentación entre el cliente y el equipo de trabajo. El desarrollo con SXP se realiza en iteraciones cortas (sprints) a lo largo de 3 fases, dándole cumplimiento a un grupo de actividades, de las que se generan una serie de artefactos que documentan el proceso de desarrollo, obteniendo un release del producto con nuevas funcionalidades [18].

## 1.5 Lenguajes, tecnologías y herramientas de desarrollo utilizadas

### 1.5.1 Lenguajes.

#### ➤ CSS

CSS, del acrónimo en inglés Cascading Style Sheets (Hojas de estilo en cascada), es un mecanismo

simple para darle estilo (ej, fuente, color, espacio) a los documentos web [22]. El W3C es el encargado de definir los estándares a utilizar por los navegadores y actualmente existen los estándares CSS1, CSS2.0, CSS2.1 y se trabaja en el estándar CSS3 ya soportado por algunos navegadores modernos (Mozilla Firefox y Safari). El uso de hojas de estilo permite el control de la presentación de la información mostrada en los sitios web y da la posibilidad de escoger el estilo de presentación de acuerdo al dispositivo que la muestre. Con la introducción de CSS, se cambió el viejo modelo de maquetación mediante tablas que le resta rendimiento a las páginas web.

## ➤ **PHP**

PHP (de las siglas en inglés Hypertext pre-procesor), es un lenguaje de programación web orientado a objeto (en su versión 5) y compilado del lado del servidor (desde la versión 4). Es posiblemente el lenguaje más usado para la programación de sitios web dinámicos por su sencillez y prestaciones además de estar publicado bajo PHP Licence v3.01 (<http://www.php.net>), considerada como software libre por la Free Software Foundation.

- Permite la conexión a diferentes servidores de base de datos, tales como MySQL, PostgreSQL, Oracle, Microsoft SQL Server, Firebird y SQLite.
- Puede ser ejecutado en la mayoría de los sistemas operativos \*NIX(Linux, MAC OS) y Windows.
- Se integra a los servidores web más utilizados, Apache e IIS(Internet Information Server).
- Es portable.
- Alto rendimiento en la ejecución de los scripts.

PHP es el heredero de un producto anterior, llamado PHP/FI, creado por Rasmus Lerdorf en 1995, concebido como un conjunto de scripts ( denominados por él “Personal Home Page Tools”), para controlar el acceso a su trabajo online. Poco a poco fueron aumentando las necesidades de gestión de su información personal por lo que fue agregándole un conjunto de librerías escritas en lenguaje C capaces de conectar con bases de datos y permitía a los usuarios escribir pequeñas aplicaciones Web dinámicas. Rasmus decide liberar el código de PHP/FI para que cualquiera pudiera utilizarlo y mejorar o arreglar posibles errores.

Inicialmente la sintaxis de las variables de PHP/FI era muy similar a la de Perl y sólo se podían ejecutar algunas de las funcionalidades básicas de las que cuenta hoy en día y ya en 1993 surge la

primera versión bajo el nombre de PHP 3 incorporando soporte para varias bases de datos, protocolos y APIs además de convertirse en un lenguaje escalable. Bajo el nombre de PHP, lo que eliminaba todo vestigio de uso personal, fue adoptado por millares de usuarios.

En mayo de 2000, fue liberado PHP 4 con un nuevo motor (Zend) con mejoras en código base haciéndolo modular y concebido para la ejecución de aplicaciones complejas. Se incluyeron nuevas características como soporte para múltiples servidores Web, manejos de sesiones de usuarios, buffers de salidas y formas más seguras de control de entradas de usuarios. En su última versión, PHP 5, se convierte en un lenguaje de programación orientado a objeto usando la nueva versión del motor Zend (versión 2) [19].

#### ➤ **XML**

XML (del inglés extensible markup language), no es como su nombre lo indica un lenguaje de marcado, sino un metalenguaje que permite definir lenguajes de marcado adecuados a usos determinados. Surge a partir del lenguaje GML (General Markup Language) ideado por la compañía IBM como necesidad en la organización de su información. La idea que subyace bajo XML es la de crear un lenguaje muy general que describa el contenido de lo que etiqueta.

XML establece una estructura u orden jerárquico para las etiquetas, obligando a la organización del código. Permite describir prácticamente cualquier tipo de dato, llamadas a procedimientos remotos, contenidos multimedia o escribir interfaces de comunicación entre aplicaciones. Su uso se extiende al intercambio de información contable, actualizaciones de software y almacenamiento de gráficos vectoriales. Es usado ampliamente en los servicios web.

#### ➤ **YAML**

YAML (del inglés, YAML Ain't Markup Language) es un lenguaje de serialización de datos [20], basado en XML, pero más sencillo, ya que no integra las etiquetas de apertura y cierre. Su uso se extiende a múltiples aplicaciones, principalmente en ficheros de configuración e intercambio ligero de información y como lenguaje para la representación del esquema de base de datos para el framework de ORM (del inglés, Object Relational Mapping) Doctrine [21].

#### ➤ **XHTML**

XHTML, del acrónimo en inglés Extensible Hypertext Markup Language (lenguaje extensible de marcado de hipertexto), es un estándar propuesto por la W3C (World Wide Web Consortium) como

una redefinición del lenguaje HTML que permite la creación de páginas web. XHTML hace más estrictas algunas de las reglas del HTML para que sea compatible con XML y posibilita el contenido web sea posible visualizarlo desde numerosos dispositivos donde la sintaxis HTML es difícil de afrontar.

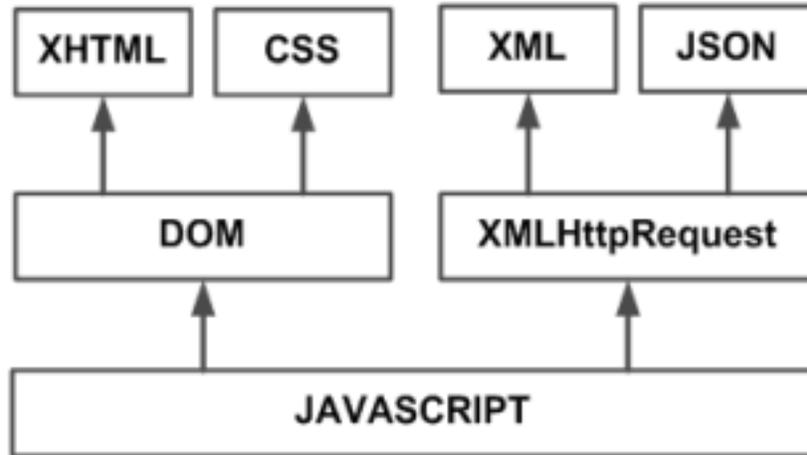
## **1.5.2 Tecnologías**

### **➤ AJAX**

El término AJAX es un acrónimo de Asynchronous JavaScript + XML, que se puede traducir como JavaScript asíncrono + XML. No constituye una tecnología en sí mismo, en realidad, se trata de varias tecnologías independientes unidas entre sí que pueden lograr en las aplicaciones web el mismo ambiente de las aplicaciones de escritorio. El término fue anunciado por primera vez en el artículo “Ajax: A New Approach to Web Applications”.

Las tecnologías que forman AJAX son:

- XHTML y CSS, para crear una presentación basada en estándares.
- DOM (Document Object Model), para la interacción y manipulación dinámica de la presentación.
- XML, XSLT y JSON, para el intercambio y la manipulación de información.
- XMLHttpRequest, para el intercambio asíncrono de información.
- JavaScript, para unir todas las demás tecnologías.[23]



*Figura 1.2: Tecnologías que se agrupan bajo el término AJAX*

AJAX permite una mejor experiencia del usuario en cuanto a la interacción con la aplicación, debido a que elimina la recarga constante de las páginas mediante un elemento intermedio entre el usuario y el servidor, que ejecuta la petición en segundo plano y no permite que el usuario se enfrente a una página vacía en respuesta del servidor web. También en respuesta del servidor web a la petición del usuario cuando presiona un enlace o botón utilizando el estándar DOM y Javascript para la manipulación de la presentación solo se verá afectada una parte de toda la página web. A partir del uso de estas tecnologías los usuarios han experimentado otra forma de crear aplicaciones interactivas que pueden sustituir a los flash y las aplicaciones escritorio.

### **1.5.2.1 Frameworks de Desarrollo**

Un framework (marco de trabajo) es un conjunto de librerías y herramientas sobre las que puede apoyarse el desarrollo de un proyecto. “Un framework proporciona estructura al código fuente, forzando al desarrollador a crear código más legible y más fácil de mantener” [24]. Establece una arquitectura mediante el uso automatizado de patrones y agiliza el desarrollo de las aplicaciones, ya que el programador no tiene que centrarse en detalles del lenguaje sino en la lógica de las mismas. Un framework encapsula operaciones complejas en instrucciones sencillas y se encarga de automatizar las tareas más comunes.

### 1.5.2.2 Frameworks para JavaScript

Con el surgimiento del HTML dinámico han proliferado una enorme cantidad de frameworks para Javascript, que se encargan de muchas características ventajosas que ofrecen dinamismo a las aplicaciones web. Algunos de los framework más utilizados en la actualidad son descritos a continuación.

#### ➤ **Prototype**

Es un framework que persigue el objetivo de crear aplicaciones web dinámicas. Se caracteriza por ser útil en las interacciones AJAX entre el servidor y el cliente, permite manejo de eventos de usuario, efectos visuales en las páginas web y manipulación del DOM para modificar las características de las páginas de manera fácil.

- Soportado por varios navegadores web.
- Es software libre, lo que posibilita el estudio y modificación de su código fuente para mejorar en bien de la comunidad de desarrollo web.

#### ➤ **ExtJS**

La librería ExtJS<sup>7</sup> comenzó siendo una extensión para la popular Yahoo User Interface (Interfaz de Usuario de Yahoo). Pasó mucho tiempo para que ExtJS tuviese desarrolladores de código abierto que contribuyeran con su conocimiento a construir lo que hoy se conoce como una de las más robustas librerías de desarrollo del lado del cliente. Su diseño y arquitectura brindan las características esenciales para obtener una librería completa para construir interfaces de usuario intuitivas, amigables y flexibles:

- Provee una serie de interfaces de usuario iguales a las de las aplicaciones de escritorio, lo que permite al desarrollador concentrarse en la funcionalidad de la aplicación.
- Puede trabajar por medio de otras librerías Javascript usando adaptadores.
- Es recomendable cuando se requiere una gran interactividad del usuario con la aplicación web, como aquellos sistemas que requieran de un flujo de trabajo complejo.
- Es compatible con los navegadores web Internet Explorer 6+ , Firefox 1.5+ (PC, Mac) , Safari 2+ , Opera 9+ (PC, Mac) .

---

<sup>7</sup> Sitio oficial: <http://www.extjs.com>

- Funcionalidades e interfaces controladas por eventos. Esto permite que en cada interacción del usuario con la aplicación ExtJS interviene en la presentación de la respuesta que da la aplicación al usuario, ya sea con intervención del servidor web (interacciones AJAX) o no [25].

### ➤ **JQuery**

JQuery<sup>8</sup> es un eficiente y ligero framework Javascript que facilita el desarrollo de aplicaciones web interactivas. Distribuido bajo licencias MIT y GPL, JQuery ofrece una excelente API que facilita el uso de las funciones del DOM así como otras no soportadas por éste, manipulación de CCS, incluye funcionalidades para el trabajo con AJAX evitando la carga del servidor , manipulación de eventos, animaciones, diseño y efectos del contenido web de forma sencilla. Es de fácil aprendizaje y posee una excelente documentación en su sitio oficial.

#### **Características:**

- Cuenta con una amplia variedad de plugins desarrollados sobre la base de la propia librería JQuery, que permiten obtener efectos sorprendentes y facilidades en un sitio web.
- Se complementa con un conjunto de interfaces de usuario prediseñadas (jQueryUI), similares a las de escritorio, que permiten aumentar la familiarización del usuario con la aplicación web.
- Es completamente funcional en los navegadores web Internet Explorer 6.0, Firefox 2 o superior, Opera 9.0 o superior, Safari 3.0 o superior y Chrome.
- Soporta selectores para CSS en sus versiones 1 a la 3.
- Es usado por las empresas de software más exitosas de los últimos tiempos en las más variadas esferas del desarrollo, por ejemplo: la organización Mozilla, Google, NBC, CBS, entre otras.
- Usado en varios proyectos de software como Drupal y Wordpress.

#### **Criterios para la selección del framework de Javascript**

Se escoge JQuery como framework de JavaScript para el desarrollo de la solución, ya que posee una amplia variedad de plugins, widgets (componentes visuales) y utilidades que permiten tener un control total de las interfaces de usuario. Además de permitir el intercambio asíncrono de datos entre cliente y servidor de manera sencilla, la librería JQueryUI construida sobre el propio framework JQuery provee un conjunto de widgets y efectos de animación avanzados que posibilitan el desarrollo

---

8 Sitio oficial: <http://jquery.com>

de aplicaciones web interactivas y con una gran experiencia de usuario. La sintaxis de código es muy fácil de aprender y sencilla de utilizar; además soporta CSS en sus versiones 1, 2 y 3.

### **1.5.2.3 Frameworks para el lenguaje PHP**

El desarrollo de aplicaciones web está actualmente muy condicionado por el uso de frameworks; si se concibe un producto medianamente grande que englobe un conjunto de funcionalidades más o menos complejas, es muy recomendable el uso de un framework. Para PHP existen un conjunto de ellos, algunos con características similares y ventajas sobre el resto, por ejemplo: CakePHP, Zend Framework, Symfony, Codelgniter, Prado, Kumbia, entre otros. Los más utilizados por los desarrolladores a nivel mundial coinciden con los siguientes: Codelgniter, CakePHP, Zend Framework, Kumbia y Symfony.

#### **➤ Codelgniter**

Codelgniter es un sencillo framework escrito para que funcione sobre PHP 4 y PHP 5 [26]. A continuación se describen un conjunto de características:

- Implementa el patrón MVC (Modelo-Vista-Controlador) y cuenta con una gran cantidad de helpers que le facilitan el trabajo a los desarrolladores.
- Implementa el patrón Active Record como capa de abstracción de base de datos, aunque no posee capa de ORM.
- Es compatible con algunos de los motores de base de datos más usados (MySQL, PostgreSQL, SQL Server, SQLite y Oracle).
- El mecanismo de control de sesiones es a través de cookies y puede usar base de datos para el control de las mismas.
- Posee una gran comunidad de desarrolladores y una basta documentación, tanto en inglés como español.

#### **➤ Kumbia**

Diseñado para acelerar el desarrollo de aplicaciones web sin pérdida de tiempo y con eficacia, con una buena documentación en español sobre su uso en Internet. Reúne el conjunto de herramientas para la automatización de tareas que quitan tiempo al desarrollador, permitiéndole invertir sus

esfuerzos en los aspectos esenciales del desarrollo. Está escrito en PHP 5 y liberado bajo la licencia GPL de software libre.

El Modelo de Objetos de Kumbia es representado en tres capas:

- Abstracción del sistema de gestión de bases de datos.
- Mapeo Objeto Relacional.
- Modelo Vista Controlador.

“Kumbia es innovador y su principal enfoque es desarrollar herramientas y escribir cada componente del framework pensando en que sea fácil de usar para cualquiera que lea su documentación” [27].

Cumple con los requisitos esenciales para construir un proyecto web:

- Soporta las mejores prácticas de diseño y patrones más eficientes.
- Multiplataforma, sencillo, fácil de aprender y configurar.
- Permite la creación de aplicaciones de código legible y fácil de mantener.
- Se integra con los sistemas gestores de bases de datos más populares en la actualidad, como por ejemplo: MySQL, PostgreSQL y Oracle.
- Administración de Caché.
- Scaffolding Avanzado.
- Implementa el patrón arquitectónico MVC.
- Brinda soporte para el uso de AJAX para asegurar una mejor interactividad y experiencia del usuario.
- Posee herramientas para la generación de formularios.
- Implementa mecanismo de seguridad para evitar accesos no autorizados a la aplicación.

Presenta herramientas que complementan su funcionalidad; tal es el caso de FPDF para los reportes en formato PDF, Prototype como framework de Javascript unido a Scriptaculous para la manipulación de la presentación y lograr los efectos visuales compatibles con varios navegadores, PHPMailer para el envío de correo electrónico y Smarty como motor de plantillas.

## ➤ **CakePHP**

Es un marco de trabajo para el desarrollo rápido y robusto de aplicaciones web, con excelentes características para lograr un producto con los requisitos que se exigen de un sistema web como su portabilidad, seguridad, reutilización de componentes, refactorización y modularización del código fuente, entre otras. Dentro de las características más significativas se relacionan:

- Automatiza un conjunto de tareas que pueden resultar molestas en algunos casos por su naturaleza trivial.
- Emplea el patrón arquitectónico MVC y accede a las bases de datos a través de una capa ORM.
- Urls amigables.
- Uso de plantillas personalizables.
- Validación de datos de entrada integrada y componentes de seguridad y sesión.
- Compatibilidad con las versiones 4 y 5 de PHP.
- Helpers integrados (fragmentos de código encapsulado en funciones PHP) para formularios, HTML, JavaScript, AJAX y otros [28].

## ➤ **Zend Framework**

Zend Framework es el framework desarrollado por la empresa que respalda comercialmente a PHP. Basado en la arquitectura MVC y escrito en PHP 5. Otros aspectos que lo caracterizan son:

- Los componentes que representan la biblioteca estándar del framework son completamente extensibles y pueden usarse individualmente por el bajo acoplamiento que mantienen.
- Zend Framework posibilita la integración con PHPUnit para el desarrollo de pruebas unitarias a la aplicación que se desarrolle con el mismo.
- Soporta diferentes motores de bases de datos, haciendo uso del patrón Active Record como capa de abstracción de base de datos; aunque no posee una capa de ORM propio, puede integrarse con el framework Doctrine.
- Contiene un conjunto de módulos para manejo de canales RSS, servicios web (Amazon, Flickr, Yahoo) y módulos de autenticación.
- Cuenta con una amplia documentación, aunque la mayoría en inglés [29].

## ➤ **Symfony**

Symfony es un framework de desarrollo web desarrollado completamente en PHP5, incorporando las características de orientación a objetos del lenguaje. Basado en la arquitectura MVC, sigue las mejores prácticas de patrones de diseño web. Entre sus principales características se destacan las siguientes:

- Fácil de usar pero a la vez tan flexible como para adaptarse a los casos más complejos.
- Esta preparado para satisfacer las necesidades de las empresas y excelente en desarrollos a largo plazo.
- Su escalabilidad lo hace extensible al integrarse con plugins y librerías desarrolladas por otras personas.
- Hace uso de Propel como capa de ORM, que facilita su integración con los motores de bases de datos relacionales más usados (MySQL, PostgreSQL, Microsoft SQL Server y Oracle), además de integrarse con Doctrine, a conveniencia del usuario.
- Symfony incluye su propio framework de pruebas (unitarias y funcionales) para la automatización de las mismas y cuenta con más de 9000 pruebas entre funcionales y unitarias sobre su mismo código, permitiendo que el mismo esté muy optimizado.

Symfony incluye un conjunto de tareas que automatizan las características de un proyecto web:

- La capa de presentación incluye plantillas de presentación globales (layouts) que constituye el enmaquetado del sistema, pudiendo diseñarlo sin necesidad de conocimientos del framework.
- Brinda una barra de depuración web para ayudar al programador a corregir errores en el código.
- Está basado en la reutilización de código a través de plugins y helpers para HTML, formularios, JavaScript y AJAX, que encapsulan una gran cantidad de código.
- Tiene incluido en su núcleo el framework Prototype, para trabajar con la capa de presentación de la aplicación, aunque se puede integrar perfectamente con JQuery, Mootools o ExtJS.
- Provee varios mecanismos de seguridad de la aplicación como son las estrategias de escape de variables para evitar ataques contra el sistema y los mecanismos de gestión de sesión y credenciales para la autenticación de los usuarios.

- Disminuye el uso de ancho de banda y la carga del servidor al gestionar la caché de la aplicación.
- El sistema de enrutamiento que incluye permite el uso de URL limpias y más tomadas en cuenta por los buscadores.

Además provee varios entornos (entorno de producción, de desarrollo y pruebas). Symfony ha sido probado en proyectos de gran envergadura y sitios como delicious.com, Yahoo Bookmarks, Yahoo Answers y Dailymotion funcionan sobre el mismo. Está bajo licencia MIT convirtiéndolo en un producto libre y cuenta con una gran comunidad de desarrolladores y documentación precisa. Se puede ejecutar en varias plataformas incluyendo los sistemas \*nix (Unix, Linux, etc) y en sistemas operativos de la familia de Windows [30].

### **Criterios para la selección del framework de php**

Se elige Symfony como framework de desarrollo para PHP porque reúne las mejores prácticas de desarrollo web e integra muchas otras librerías y herramientas que aportan fortaleza al producto final. Usa además Doctrine o Propel como ORM a opción del desarrollador, para el mapeo relacional de objetos, que constituyen los dos mejores en esta área.

Una de las novedades introducidas a partir de la versión 1.2 de Symfony lo constituyen los formularios; relacionados uno a uno con cada tabla del modelo de datos, por lo que es posible representar un formulario que realice la lógica de insertar o modificar registros de una tabla determinada, sin necesidad de escribir el código XHTML de los mismos. Este esquema de trabajo no rompe con la separación de la lógica de los datos y su representación visual. Además ofrece un conjunto de reglas de validación del lado del servidor encaminadas a asegurar que los datos de entrada se corresponden con los permitidos en cada uno de los campos de la tabla de base de datos que representa el formulario. Aprovechar esta bondad de Symfony es muy importante para el desarrollo ágil de la solución propuesta, teniendo en cuenta que será una aplicación de gestión.

Unido a lo plasmado anteriormente, las herramientas de generación de código, la interfaz de línea de comandos para la instalación del sistema desarrollado y otras tareas comunes automatizadas así como la amplia documentación oficial que posee; convierten a Symfony en uno de los mejores framework para PHP actuales. Cuenta con una amplia comunidad de desarrolladores que constituye la principal fuente de retroalimentación para el desarrollo de versiones posteriores, en las que se corrigen errores en las versiones anteriores y se agregan nuevas funcionalidades, además de seguir

una política LTS (Long Term Support), por lo que las versiones estables se mantienen durante un período de 3 años con una continua corrección de errores.

#### **1.5.2.4 Sistemas gestores de bases de datos**

##### **➤ MySQL Server**

MySQL un servidor de bases de datos relacional desarrollado por la compañía MySQL AB, subsidiaria de Sun Microsystem y esta a su vez de Oracle desde el año 2009. MySQL está bajo licenciamiento dual; licencia GNU GPL para desarrollo sin fines lucrativos y se necesita de una licencia comercial para desarrollar productos con fines comerciales. Es muy utilizado en las aplicaciones orientadas a la web por su rapidez de respuesta y facilidades de uso. Dentro de sus características figuran:

- Escrito en C y en C++ .
- Probado con un amplio rango de compiladores diferentes.
- Funciona en diferentes plataformas (Linux, MAC OS, Sun Solaris, Windows).
- APIs disponibles para C, C++, Eiffel, Java, Perl, PHP, Python, Ruby, y Tcl.
- Uso completo de multi-threaded mediante threads del kernel. Pueden usarse fácilmente múltiples CPUs si están disponibles.
- Proporciona sistemas de almacenamiento transaccionales y no transaccionales, posibilitando la adaptación del SGBD a casos concretos.
- Esquema de almacenamiento MyISAM, MEMORY, InnoDB, Archive, Federated y NDB.
- Relativamente sencillo de añadir otro sistema de almacenamiento.
- Un sistema de reserva de memoria muy rápido basado en threads.
- Velocidad de respuesta destacable.
- Tablas hash en memoria, que son usadas como tablas temporales.
- Administración basada en un esquema de usuarios y privilegios.
- Las funciones SQL están implementadas usando una librería altamente optimizada y deben ser tan rápidas como sea posible. Normalmente no hay reserva de memoria tras toda la inicialización

para consultas.

- El servidor está disponible como un programa separado para usar en un entorno de red cliente/servidor. También está disponible como biblioteca y puede ser incrustado en aplicaciones autónomas. Dichas aplicaciones pueden usarse por sí mismas o en entornos donde no hay red disponible.
- Las opciones de conexión abarcan TCP/IP, sockets y ODBC.
- Soporta los diversos tipos de columnas del estándar SQL así como tipos espaciales OpenGIS.

### ➤ **PostgreSQL**

PostgreSQL, inicialmente llamada Postgres, fue creado por Michael Stonebraker, profesor de la Universidad de Berkeley e integrante de Informix Corporation, como un proyecto predecesor de Ingres. Desarrollado entre 1986 y 1994, marcó el inicio de las tecnologías objeto-relacional dentro de las bases de datos. Distribuido bajo licencia BSD. En sus más de quince años de desarrollo activo ha incorporado un conjunto de mejoras que lo convierten en un sistema de base de datos con una arquitectura probada, que se ha ganado una sólida reputación por su fiabilidad e integridad de los datos [31]. Dentro de sus características se destacan:

- Corre sobre la mayoría de los sistemas operativos, incluyendo Linux, UNIX((AIX, BSD, HP-UX, SGI IRIX, Mac OS X, Solaris, Tru64) y Windows.
- Conjunto de APIs para diferentes lenguajes (PHP, C/C++, Java, Python, Perl, Ruby).
- Lenguaje procedural propio denominado PL/PgSQL.
- Soporte para integridad referencial.
- Soporte nativo para SSL.
- Tipos de datos del estándar SQL así como soporte para tipos de datos creados por el usuario.
- Soporte para transacciones y control de concurrencia.
- Soporte multiusuario.
- Herencia entre tablas.

Sin embargo, PostgreSQL es un poco lento con respecto a MySQL en operaciones de lectura sobre las bases de datos, debido a que siempre tiene en cuenta una característica deseable en las bases

de datos: la integridad referencial, gracias a la que se garantiza que una entidad (fila o registro) siempre se relaciona con otras entidades que existen en la base de datos. Esto implica que en todo momento dichos datos sean correctos, que no existan repeticiones innecesarias, o relaciones mal resueltas.

### **Criterios para la selección del servidor de bases de datos**

Se escoge MySQL como sistema gestor de base de datos con el motor de almacenamiento MyISAM para el desarrollo de la solución propuesta, basando la selección en la velocidad de respuesta de las consultas SQL de lectura, al obviar la integridad referencial en las tablas durante la realización de operaciones sobre las mismas; además su desarrollo está orientado a ganar en velocidad en procesadores de bajas prestaciones. MySQL constituye la mejor solución para las aplicaciones web dinámicas donde la rapidez en la presentación de la información es crucial. El sistema propuesto gestionará la información de un portal web en el que la mayoría del contenido publicado constituye información almacenada en base de datos, con poca concurrencia en la modificación de la información y las tasas de acceso concurrencio a la información oscila entre 1400 y 1800 usuarios conectados.

#### **1.5.2.5 Servidores de Aplicaciones Web**

##### **➤ Apache**

Apache surge a partir de un servidor web de la NCSA (National Center for Supercomputing Applications de la Universidad de Illinois) como corrección a errores y mejoras importantes al producto inicial. Por los resultados logrados, es fundado el Grupo Apache, potenciador del nuevo producto al que se le hicieron mejoras en su arquitectura logrando modularidad. En 1999 los desarrolladores fundan la Apache Software Foundation, organización que da soporte legal al desarrollo de Apache y que lo mantiene actualmente bajo la licencia Apache Licence 2.0, que lo convierte en software libre [32]. Apache, en su versión 2, hace del servidor web una solución más flexible y escalable, ofreciendo un conjunto de mejoras. Dentro de sus características están:

- Soporte para el protocolo HTTP 1.1.
- Sencillo, con la configuración basada en archivos fáciles de configurar.
- Soporte para CGI (Common Gateway Interface).

- Soporte para FastCGI.
- Soporte de Host Virtuales.
- Soporte de autenticación HTTP.
- Perl integrado.
- Soporte de scripts PHP.
- Soporte de servlets de Java.
- Servidor proxy integrado.
- Estado del servidor y adaptación de registros.
- Soporte de Server Side Include (SSI).
- Soporte de Secured Socket Layer (SSL).
- Módulos multiproceso.
- Portable en tiempo de ejecución [33].

➤ **Internet Information Server (IIS)<sup>9</sup>**

IIS es uno de los productos cumbres de la empresa Microsoft; engloba un conjunto de herramientas destinadas al servicio de Internet (servidor Web, servidor FTP, servidor de correo y servidor de noticias). Dentro de sus características se destaca:

- Aprovecha las últimas tecnologías de cifrado y métodos de autenticación (SSL).
- Autenticación a través de servidores proxy.
- Control de acceso a través de listas de restricciones de direcciones IP.
- Fácil instalación a través de asistentes de interfaz gráfica.
- Seguro y administrable.
- Confiable y escalable.
- Soporte nativo para lenguaje ASP.
- Permite integración con PHP.

---

<sup>9</sup> <http://www.iis.net>

- Integrado como paquete adicional a los sistemas operativos Windows XP, Windows NT, Windows 2000.
- Software privativo.

### **Criterio para la selección del servidor Web**

Para darle solución al problema planteado en la presente investigación se escoge como servidor web a Apache en su versión 2.2.15, por las características favorables en cuanto a rendimiento, seguridad y escalabilidad. Como estrategia de la Universidad de las Ciencias Informáticas de uso de software libre en sus sistemas, Apache constituye la alternativa libre en el hospedaje de aplicaciones para la web, con soporte nativo para PHP, lenguaje también seleccionado para el desarrollo de la solución propuesta. Es el servidor web por excelencia para sistemas operativos GNU/Linux.

#### **1.5.2.6 Graficación de estadísticas**

Una de las maneras de representar la información visual en las aplicaciones web es a través de gráficos que muestran dicha información en diferentes formas de apreciación, siendo de gran utilidad en los procesos de análisis estadístico, presentándola de una manera más sistemática y resumida. A continuación se describen dos librerías usadas con este fin: HighCharts y Open Flash Chart.

##### **➤ Librería HighCharts<sup>10</sup>**

Es una librería para la graficación en la web escrita completamente en JavaScript, que no requiere complementos adicionales instalados en los navegadores para realizar su funcionamiento, tales como: Flash o Java. Tampoco requiere tecnologías de lado del servidor instaladas como PHP o ASP.NET. Permite mostrar los datos auxiliándose de diversos tipos de gráficos con excelente calidad visual; entre ellos los gráficos de pastel, líneas, barras, columnas, entre otros. Además permite combinaciones de estos tipos de gráficos en un solo gráfico. Entre las características que destacan a esta librería se encuentran:

- Es compatible en muchos navegadores como los de los dispositivos iPhone o Internet Explorer versión 6 o superior, además de Mozilla Firefox, Opera, Google Chrome y otros.
- Posee licenciamiento dual, es decir, por una parte es software libre para uso no comercial, y se requiere una licencia para usarlo en aplicaciones web que serán comercializadas.

---

<sup>10</sup> <http://www.highcharts.com>

- La obtención de datos de manera asíncrona a través de las funciones AJAX de los frameworks JQuery o MooTools hacen que la graficación sea dinámica, actualizándose los valores periódicamente con datos enviados desde el servidor u otra fuente de datos.
- Soporta múltiples ejes coordinados, mensajes con determinada información que son mostrados al señalar con el mouse determinada zona del gráfico (tooltips), gráficos invertidos o ejes coordinados invertidos, rotación de los textos de las etiquetas en los ejes coordinados, así como examinar especialmente determinada zona del gráfico de interés (zoom).

#### ➤ **Librería Open Flash Chart (OFC)**

Es una librería para la graficación en la web auxiliándose del formato flash en la representación de los datos. El proceso consiste en embeber en una página web una película flash en formato swf, a la cual se le asigna como parámetros la fuente de datos que brindará la información a mostrar en formato JSON<sup>11</sup> para ser generado el gráfico posteriormente, y el elemento de la página donde será embebido el gráfico generado. Esta librería aprovecha la relevante calidad visual del formato flash, ofreciendo una experiencia al usuario agradable.

#### **Características:**

- Permite la representación de los datos en múltiples tipos de gráficos, por ejemplo: de barras (en dos dimensiones y tres dimensiones), columnas, líneas, pastel, de área bajo la curva, y combinaciones de ellos.
- Es posible generar una imagen del gráfico representado utilizando la propia librería.
- Permite hacer clic sobre las barras o las líneas y levantar un mensaje brindando determinada información.
- Los datos pueden ser generados a través de implementaciones realizadas en diferentes lenguajes como PHP, Java, .NET, Perl, Python y Ruby.

#### **Desventajas:**

- El uso del formato flash en la web demora la carga de la página.
- Requiere del complemento flash en el navegador.

---

<sup>11</sup> <http://www.json.org>

- La liberación del códec de vídeo VP8 por la compañía Google y la inclusión de éste en HTML 5, hará que la tecnología flash quede obsoleta, ofreciendo mayor velocidad de carga de las páginas web con respecto al formato flash.

### **Criterio de selección de la librería para la graficación**

Para darle solución al problema planteado en la investigación se selecciona la librería HighCharts versión 1.2.5 para la representación gráfica de los datos referidos a las encuestas de los programas grabados que se publican en el portal Inter-Nos. La librería está escrita completamente en JavaScript, una tecnología estándar de la web que permite la representación de la información mediante la manipulación de los elementos que componen las páginas web.

Para su funcionamiento no requiere complementos adicionales instalados en los navegadores web como Java o Flash, y además es muy fácil configurarla. También es relevante destacar el uso del formato JSON para el intercambio ligero de datos, así como la generación de los objetos que componen el gráfico resultante, aspectos que agilizan la respuesta a la petición del usuario delegando el proceso de creación del gráfico a las máquinas clientes y liberando al servidor de esta tarea; ventajas que no presenta el uso de la librería Open Flash Chart.

### **1.5.2.7 Servicios Web**

Desde el mismo momento en que las redes son el medio de trabajo de muchas aplicaciones, los creadores y desarrolladores de software pensaron en compartir tareas entre ellas. A causa de la imperiosa necesidad de que las aplicaciones se comunicaran a través de Internet es que surge la denominada Arquitectura Orientada a Servicios (SOA, de sus siglas en inglés Services Oriented Architecture), que elimina la pérdida de tiempo, la corrupción de datos y el gran costo que se incurre en muchas ocasiones que se desea integrar varios sistemas.

Hoy en día la integración de sistemas utilizando SOA brinda ventajas en cuanto a la reusabilidad de funcionalidades, independencia e interoperabilidad de los sistemas involucrados. SOA viene a ser la evolución de aplicaciones monolíticas y cerradas a una arquitectura en capas de abstracción compuesta por servicios web [34].

### **¿Qué es un servicio web?**

Un servicio web es un sistema software diseñado para soportar la interacción máquina-máquina a

través de una red, utilizando protocolos estándares de Internet y XML como formato de mensajes. Los servicios web no están orientados a sistema operativo alguno o lenguaje de programación, son autodescriptivos por medio de una gramática XML y proveen un mecanismo simple para su publicación y utilización.

La arquitectura de un servicio web puede ser definida de la siguiente manera:

- **Service Transport:** responsable de transportar los mensajes entre las aplicaciones a través de protocolos de Internet, ejemplo HTTP, SMTP, FTP y POP.
- **XML Messaging:** capa responsable de la codificación de los mensajes en un formato XML común, de modo que los mensajes puedan ser entendidos por las aplicaciones que se comunican. Actualmente esta capa incluye las variantes XML-RPC, REST, WDDX y SOAP.
- **Service Description:** capa responsable de la descripción de la interfaz pública del servicio web.
- **Service Discovery:** capa responsable de la centralización de los servicios en un registro común y permite la fácil publicación y/o búsqueda de los mismos.

### **Lenguaje de Descripción del Servicio Web (WSDL)**

Es una gramática XML usada para describir servicios web. Define cómo el servicio web es accedido y las operaciones que ejecuta, cómo son pasados los mensajes entre las aplicaciones y la estructura de los mismos [35]. Permite que del lado del consumidor no sea necesario conocimiento alguno de la API del servicio web, mientras que del lado del creador del servicio puede ser un poco engorroso la conformación del documento; aunque existen herramientas que automatizan esta tarea. El documento WSDL no es requerido para crear un servicio web, sin embargo es una parte esencial del WS-IBP (Perfil Básico de la Organización de Interoperabilidad de Servicios Web) <sup>12</sup>.

### **UDDI (Universal Description, Discovery, and Integration)<sup>13</sup>**

UDDI es un estándar básico para proveer un registro universal, en el que los servicios web pudiesen ser localizados, documentados a través de WSDL y consumidos. Considerado uno de los factores para que un servicio web esté completo [36].

---

<sup>12</sup> <http://www.ws-i.org> (Web Services Interoperability Organization)

<sup>13</sup> <http://www.uddi.org>

## Variantes de la capa XML Messaging

### ➤ **WDDX (Web Distributed Data Exchange)**

Es una tecnología basada en XML que permite el intercambio de datos y estructuras de datos entre sistemas en un formato neutral. Se limita a definir tipos de datos simples (entero, string, boolean, float, etc) y tipos de datos completos solo arreglos y recordsets<sup>14</sup>. No es considerado un servicio web pero puede ser usado para crearlo. Tampoco es un estándar formal, pero está construido sobre el estándar XML 1.0, aunque no incluye un mecanismo de transporte de datos entre sistemas.

### ➤ **XML-RPC (Llamada a procedimientos remotos sobre HTTP vía XML)**

Es considerado el punto de inicio de los servicios web. Consiste en la llamada a un procedimiento o función que se ejecuta en una máquina remota vía XML, para el intercambio de parámetros de la función y los resultados obtenidos. No solo especifica el formato del mensaje XML, sino que también define como será transportado el mensaje (peticiones HTTP POST por defecto). Soporta la definición de los tipos de datos simples al igual que WDDX, así como arreglos y estructuras como tipos de datos complejos. XML-RPC no estandariza los códigos de error de las peticiones.

### ➤ **REST (Representational State Transfer)**

No es un estándar específico como otras tecnologías para servicios web, más bien es un estilo arquitectónico que utiliza protocolos y tecnologías existentes a las que los desarrolladores están acostumbrados, razón por la cual es muy usado. Utiliza HTTP como protocolo de comunicación, aunque no define un estándar XML para el envío de los mensajes (no define un contrato).

Un servicio web basado en REST retorna un XML con una estructura definida por el implementador del servicio. Con REST se obtienen mejores tiempos de respuestas respecto a otras tecnologías y una disminución considerable en la carga del servidor. Al aplicarlo a los servicios web, defiende el uso estricto del protocolo HTTP. Es sencillo de usar desde la perspectiva del consumidor.

### ➤ **SOAP**

Cuando surge SOAP en su versión 1.1 se le denominó por el acrónimo en inglés Simple Object Access Protocol, pero en la versión 1.2 (actual) es simplemente conocido por SOAP. Es un protocolo ligero para el intercambio de información en entornos distribuidos y descentralizados. PHP desde su versión 5.0.\* tiene soporte nativo para SOAP a través de una extensión. Desde la perspectiva del

---

14 Conjunto de registros

creador del servicio es más complejo que desde la perspectiva del consumidor.

SOAP auxiliándose del WSDL adquiere ventajas sobre REST y XML-RPC, al especificar un contrato entre el cliente y el proveedor del servicio. Cuando se desea implementar un servicio web desde un lenguaje de programación orientado a objetos, SOAP aprovecha las características de los tipos de datos complejos. Un mensaje SOAP, es un documento XML, por lo que la seguridad de los datos enviados, puede asegurarse a través de un conjunto de técnicas: Encriptación XML, Firma Digital XML, Certificados XKMS (XML Key Management Specification), SAML (Security Assertion Mark-up Language ) y validación de datos. Integrado en lenguajes como PHP, JAVA y C# a través de APIs [37].

### **Selección de tecnologías para implementar los servicios web**

De acuerdo al estudio realizado sobre las tecnologías relacionadas con los servicios web y en concordancia con los lineamientos de arquitectura establecidos para la intranet de la Universidad de las Ciencias Informáticas [38], se escoge SOAP como capa de mensajería para la implementación de los servicios web, por las ventajas que ofrece en relación a la descripción del servicio, auxiliándose de un documento WSDL. El desarrollo de aplicaciones que proveen servicios web con el lenguaje PHP, específicamente con el framework Symfony, toman ventajas del plugin ckWebServicePlugin, que aprovecha la capa de acceso a datos del framework y expone las acciones de la capa correspondiente al controlador provista por la arquitectura del framework como servicios web, haciendo uso de SOAP; así como la generación automática del documento WSDL que cumple con los lineamientos de la WS-I, de máxima interoperabilidad con clientes PHP, .NET y Java.

Las principales razón por la que no se escoge REST radica en que no establece un contrato entre las aplicaciones que se comunican, es el implementador del servicio web quien define la estructura del mensaje XML, teniendo como consecuencia que sea difícil de entender si no se escribe en una manera clara. Por otro lado, cuando se desea enviar en el mensaje XML tipos de datos complejos, REST utiliza estructuras, que suelen ser poco entendibles.

### **1.5.3 Herramientas de soporte al proceso de desarrollo de software**

#### **➤ Netbeans IDE<sup>15</sup>**

Netbeans es un IDE (del inglés, Integrated Development Environment) gratuito de código abierto, con soporte para PHP y características visuales para el desarrollo web. Se integra muy bien con sistemas

---

<sup>15</sup> <http://www.netbeans.org/>

de control de versiones y sistema gestores de bases de datos (MySQL y PostgreSQL). Funciona sobre Linux, MAC OS, Solaris y Windows. En su versión 6.8, integra soporte para el framework Symfony versión 1.2.

➤ **phpMyAdmin**<sup>16</sup>

Es una herramienta libre escrita en PHP con el objetivo de administrar servidores MySQL a través de la web. Soporta un amplio rango de operaciones sobre las bases de datos, como la administración de bases de datos, tablas, registros, usuarios y permisos; ejecución de consultas SQL e importación y exportación de datos en múltiples formatos. Disponible en 58 idiomas y con una amplia documentación.

➤ **Firebug**<sup>17</sup>

Firebug es un complemento para el navegador Mozilla Firefox que permite inspeccionar y editar el código Javascript, XHTML y CSS de una página web así como analizar las peticiones y respuestas del cliente y servidor, monitorizar el tiempo de carga de las páginas, además de analizar las peticiones vía Ajax. Está distribuido bajo licencia GPL.

➤ **Pencil**

Pencil es un plugin libre para Mozilla Firefox, distribuido bajo licencia GPL versión 2 y disponible para todas las plataformas que soporten Mozilla Firefox (hasta versión 3.\*); permite el diseño de prototipos de interfaces de usuario de manera sencilla y rápida.

➤ **Visual Paradigm for UML**<sup>18</sup>

Visual Paradigm for UML 6.4 es una herramienta CASE (Computer Aided Software Engineering, Ingeniería de Software Asistida por Ordenador) para el modelado visual, generación de documentación y de código a través de la plataforma JAVA, que soporta todo el ciclo de vida de un proyecto de software. Se integra con los principales IDEs (Eclipse, Netbeans, Jdeveloper, etc). Además de dar soporte para el modelado UML y procesos de negocio, provee el ORM para los lenguajes de programación Java, .NET y PHP. Al estar implementado sobre plataforma Java lo convierte en un producto multiplataforma. El paquete está disponible en las versiones empresarial y

---

16 [http://www.phpmyadmin.net/home\\_page/index.php](http://www.phpmyadmin.net/home_page/index.php)

17 <http://www.getfirebug.com>

18 <http://www.visual-paradigm.com>

comunitaria; la última no necesita pago de licencia alguna.

➤ **RapidSVN<sup>19</sup>**

RapidSVN es un cliente de interfaz gráfica para la comunicación con servidores Subversion. Facilita el versionado de ficheros, desde una interfaz sencilla e intuitiva. Está disponible para plataformas Windows, Linux, MAC OS y Solaris. Es rápido y eficiente, está escrito en C++. Distribuido bajo licencia GPL.

## **1.6 Conclusiones parciales**

En el desarrollo del Capítulo 1 se trataron conceptos generales acerca de la gestión de la información y sistemas de gestión de información online, con el objetivo de ubicar al lector en el dominio del problema. Son analizados algunos sistemas de gestión de contenidos audiovisuales para documentar características y funcionalidades que proveen y determinándose que es necesaria la implementación de un sistema web que esté acorde con las necesidades de gestión de la información publicada en el portal Inter-Nos.

Para ello se abordan elementos de la metodología SXP que guiará el proceso de desarrollo de software, así como las principales tecnologías y lenguajes usados en la construcción de aplicaciones web dinámicas, apostando por el conjunto integrante de la plataforma LAMP (Linux, Apache, MySQL, PHP/Python/Perl) por constituir un entorno de desarrollo de software estable, seleccionando Apache 2.2.12, MySQL 5.1.45, Symfony 1.2.4 y JQuery 1.3.2 para el desarrollo, y como Sistema Operativo la distribución GNU/Linux Debian 5.0, por las libertades de software ofrecidas.

Se describen conceptos relacionados con los servicios web, eligiéndose para la implementación de los mismos a SOAP como capa de mensajería en conjunto con XML para la codificación de los mensajes y WSDL para su descripción. Se definen además un conjunto de herramientas de soporte al ciclo de desarrollo.

---

<sup>19</sup> <http://www.rapidsvn.org/>

## Capítulo 2 Definición del sistema propuesto

En el presente capítulo se recogen las funcionalidades de la solución propuesta reflejadas en las Historias de Usuarios con sus prototipos de interfaces de usuario no funcionales, así como la planificación de las iteraciones del ciclo de desarrollo.

### 2.1 Lista de Reserva del Producto (LRP)

La lista de reserva del producto es una colección organizada y priorizada de los requisitos sobre el producto. El objetivo de la misma radica en cubrir las cualidades requeridas en el software y determinar el orden en que se le irá dando cumplimiento durante las **iteraciones o sprint**<sup>20</sup> a cada requerimiento recogido según la prioridad establecida en cada uno; también abarca las características que el producto debe tener. A continuación se recogen en cada una de las filas de la tabla mostrada los elementos que describen cada requisito identificado. Se recoge el nombre del desarrollador que dará cumplimiento al requisito, el **número identificativo del requisito**<sup>21</sup>, la descripción del requisito, el tiempo estimado para su cumplimiento y el nombre del estimador respectivamente.

Asignado a	Ítem *	Descripción	Estimación	Estimado por
		Prioridad		Muy Alta
Francisco	9.1.1	Insertar protocolo de transmisión de las medias	5 horas	Sergio
	9.2.1	Modificar protocolo de transmisión de media	5 horas	Sergio
	9.3.1	Eliminar protocolos de transmisión de medias[uno o varios a la vez]	3 horas	Sergio
	9.4.1	Establecer uno de los protocolos existentes como seleccionado por defecto	1 horas	Sergio
	9.5.1	Listar protocolos de	1 hora	Sergio

<sup>20</sup> Ciclo iterativo o período de tiempo en el que se implementa o mejora una funcionalidad del sistema. Durante un sprint el producto puede ser diseñado, implementado y probado para producir nuevos incrementos.

<sup>21</sup> **Notación HU.R.I** : significa la historia de usuario HU que dará cumplimiento al requisito R en la iteración o sprint I.

		transmisión de medias		
Javier	1.1.2	Insertar fecha de cartelera.	1 día	Sergio
	1.2.2	Eliminar fecha de cartelera [uno o varios a la misma vez]	1 día	Sergio
	1.3.2	Modificar fecha de cartelera	1 día	Sergio
	1.4.2	Listar fechas de carteleras	1 día	Sergio
	1.5.2	Insertar Canal	1 día	Sergio
	1.6.2	Modificar Canal	1 día	Sergio
	1.7.2	Eliminar Canal [uno o varios a la misma vez]	2 días	Sergio
	1.8.2	Listar canales	1 día	Sergio
	1.9.2	Insertar Cartelera	2 días	Sergio
	1.10.2	Modificar Cartelera	1 día	Sergio
	1.11.2	Eliminar Cartelera	1 día	Sergio
	1.12.2	Listar Carteleras	1 día	Sergio
Javier	2.1.2	Insertar secciones de películas	1 día	Sergio
	2.2.2	Modificar secciones de películas	1 día	Sergio
	2.3.2	Eliminar secciones de películas[uno o varios a la misma vez]	3 días	Sergio
	2.4.2	Listar secciones de películas	1 día	Sergio
	2.5.2	Insertar películas por secciones[ asociar a la película calendario de publicación sin periodicidad y protocolo de transmisión ]	4 días	Sergio
	2.6.2	Modificar películas	2 días	Sergio
	2.7.2	Eliminar películas[uno o varios a la misma vez]	2 días	Sergio
	2.8.2	Listar películas por secciones	1 día	Sergio
Javier	3.1.1	Todas las medias tienen un calendario de publicación asociado, que puede ser con periodicidad o sin	1 semana	Sergio

		periodicidad. El calendario no se autogestiona, la responsabilidad de crearlo, modificarlo o eliminarlo es de los materiales que lo requieran.		
	3.2.1	Permitir copiar un calendario existente a la hora de crear una media	1 semana	Sergio
Francisco	4.1.1	Insertar sección de series	1 hora	Sergio
	4.2.1	Modificar sección de series	1 hora	Sergio
	4.3.1	Eliminar sección de series[uno o varios a la misma vez]	4 horas	Sergio
	4.4.1	Listar secciones [ todas o solo las publicadas]	4 horas	Sergio
	4.6.1	Forzar publicación de una sección de series	2 horas	Sergio
	4.7.1	Insertar serie por sección	4 horas	Sergio
	4.8.1	Modificar serie por sección	4 horas	Sergio
	4.9.1	Eliminar serie[uno o varios a la misma vez]	3 horas	Sergio
	4.10.1	Forzar publicación de la serie	2 horas	Sergio
	4.11.1	Listar series por sección	4 horas	Sergio
	4.12.1	Insertar temporadas de series	3 horas	Sergio
	4.13.1	Modificar temporadas de series	3 horas	Sergio
	4.14.1	Eliminar temporadas de series[uno o varios a la misma vez]	2 horas	Sergio
	4.15.1	Listar temporadas de series	3 horas	Sergio
	4.16.1	Insertar capítulos por temporadas[ asociar al capítulo calendario de	4 días	Sergio

		publicación sin periodicidad y protocolo de transmisión ]		
	4.17.1	Modificar capítulos de temporadas	4 días	Sergio
	4.18.1	Eliminar capítulos de una temporada[uno o varios a la vez]	4 horas	Sergio
	4.19.1	Listar capítulos por temporadas de serie	3 horas	Sergio
		<b>Prioridad</b>	<b>Alta</b>	
Alexander	6.1.3	Insertar secciones de programas grabados	1 hora	Sergio
	6.2.3	Modificar secciones de programas grabados	1 hora	Sergio
	6.3.3	Eliminar secciones de programas grabados[uno o varios a la vez]	1 hora	Sergio
	6.4.3	Forzar publicación programas grabados	1 día	Sergio
	6.5.3	Listar secciones de programas grabados[listar todas o solo las publicadas]	4 horas	Sergio
	6.7.3	Insertar programa grabado por sección	1 día	Sergio
	6.8.3	Modificar programas grabados	4 horas	Sergio
	6.9.3	Eliminar programa grabado[uno o varios a la misma vez]	1 hora	Sergio
	6.10.3	Listar programas grabados por secciones	1 día	Sergio
	6.11.3	Forzar publicación de programas grabados	2 horas	Sergio
	6.12.3	Insertar emisión para un programa grabado[asociar	1 día	Sergio

		calendario de publicación sin periodicidad y protocolo de transmisión]		
	6.13.3	Modificar emisión de un programa grabado	2 horas	Sergio
	6.14.3	Eliminar emisión de programas grabados[uno o varios a la vez]	1 hora	Sergio
	6.15.3	Listar emisión de programa grabado	2 días	Sergio
Alexander	7.1.3	Insertar sección de radio o televisión en vivo	1 hora	Sergio
	7.2.3	Modificar sección de radio o televisión	1 hora	Sergio
	7.3.3	Eliminar sección de radio o televisión[una o varias a la vez]	1 hora	Sergio
	7.4.3	Listar secciones de radio y televisión[listar todas o solo las publicadas]	1 día	Sergio
	7.5.3	Forzar publicación de la sección de radio o televisión	1 día	Sergio
	7.7.3	Insertar estación de radio o televisión[asociar uno o varios calendarios de publicación con o sin periodicidad]	2 días	Sergio
	7.8.3	Modificar estación de radio o televisión	2 días	Sergio
	7.9.3	Eliminar estación de radio o televisión [una o varias a la vez]	2 horas	Sergio
	7.10.3	Listar estaciones por sección	1 día	Sergio
Javier	8.1.4	Insertar períodos docentes	1 hora	Sergio
	8.2.4	Modificar períodos docentes	1 hora	Sergio

	8.3.4	Eliminar períodos docente[uno o varios a la vez]	3 horas	Sergio
	8.4.4	Listar períodos docentes	1 día	Sergio
	8.5.4	Insertar Año docente	1 hora	Sergio
	8.6.4	Modificar años docentes	1 hora	Sergio
	8.7.4	Eliminar años docentes[uno o varios a la vez]	2 horas	Sergio
	8.8.4	Listar años docentes[todos o por períodos docente]	1 día	Sergio
	8.9.4	Insertar asignatura por año y período docente	2 horas	Sergio
	8.10.4	Modificar asignatura	1 hora	Sergio
	8.11.4	Eliminar asignatura[una o varias a la vez]	2 horas	Sergio
	8.12.4	Listar asignaturas por año y período docente	1 día	Sergio
	8.13.4	Insertar tipos de materiales docentes	1 hora	Sergio
	8.14.4	Modificar tipos de materiales docentes	1 hora	Sergio
	8.15.4	Eliminar tipos de materiales docentes[uno o varios a la vez]	1 hora	Sergio
	8.16.4	Listar todos los tipos de materiales docentes existentes	1 hora	Sergio
Prioridad			Media	
Francisco	4.5.5	Cambiar orden de visualización de las secciones de series publicadas	2 días	Sergio
Alexander	6.16.5	Cambiar orden de visualización de las secciones de programas	1 días	Sergio

		grabados publicadas		
	6.17.5	Insertar encuestas[se incluyen todos los programas grabados]	1 día	Sergio
	6.18.5	Modificar encuesta[incluir o quitar programas]	1 día	Sergio
	6.19.5	Eliminar encuesta[una o varia]	1 hora	Sergio
	6.20.5	Visualizar resultados de la encuesta de un mes determinado	5 días	Sergio
	7.6.5	Cambiar orden de visualización de las secciones radio y televisión publicadas	1 día	Sergio
Javier	10.1.5	Insertar usuarios del dominio UCI.CU o locales al sistema	1 día	Sergio
	10.2.5	Eliminar usuarios del sistema[uno o varios a la vez]	2 horas	Sergio
	10.3.5	Modificar usuarios del sistema	1 día	Sergio
	10.4.5	Agregar grupos de usuarios del sistema[asignar permisos al grupo]	2 día	Sergio
	10.5.5	Modificar grupo de usuarios del sistema [modificar permisos]	2 días	Sergio
	10.6.5	Eliminar grupos de usuarios del sistema[uno o varios a la vez]	2 horas	Sergio
	10.7.5	Modificar solo nombre y descripción de los permisos[Los requisitos no	1 hora	Sergio

		pueden ser eliminados]		
	10.8.5	Autenticar usuarios del sistema	2 días	Sergio
Prioridad			Baja	
Alexander y Francisco	11.1.5	Brindar servicios web que proveen la información de los materiales publicados	1 semana	Sergio
RNF (Requisitos No Funcionales)				
		LDAP funcional en la UCI		Sergio

## 2.2 Lista de Historias de Usuario

Las historias de usuario son la técnica utilizada para especificar las funcionalidades del sistema, siendo una similitud a la descripción de los casos de uso en la metodología RUP. Son esenciales en el proceso de pruebas realizadas al sistema, generándose más de un caso de prueba a realizarse por cada historia de usuario definida, para comprobar que ha sido correctamente desarrollada.

### 2.2.1 Historia de Usuario Gestionar Series

Historia de Usuario	
Número: 04	Nombre Historia de Usuario: Gestionar Series
Modificación de Historia de Usuario Número: 3	
Usuario: Francisco	Iteración Asignada: Iteración 1
Prioridad en Negocio: Muy alta	Puntos Estimados: 3 semanas
Riesgo en Desarrollo: Alto	Puntos Reales: 2 semanas y 2 días
Descripción: Esta sección garantiza toda la gestión de la publicación de series. Insertar Sección de series. Insertar serie. Insertar temporada. Insertar Capítulo. Modificar todos. Eliminar todos.	
Observaciones:	
<ol style="list-style-type: none"> <li>1. Sección de series: Categoría que agrupa a un conjunto de series. Ejemplo: Series Nacionales.</li> <li>2. Temporada: agrupa un grupo de capítulos.</li> <li>3. Los capítulos tienen: # de capítulo y título.</li> <li>4. Opcional para pensar: Sinopsis.</li> </ol> <p>Debe permitirse cambiar el orden de publicación de las secciones de series.  Debe permitir forzar la publicación de una sección de series aunque esta no tenga series publicadas.  Debe permitir cambiar el orden de visualización de las secciones publicadas.</p>	

**Prototipo de interfaz**

Título de la sección:

Forzar publicación

Limpiar Cancelar Registrar

*Figura 2.1: Crear/Modificar Sección de Series*

**Prototipo de interfaz**

Sección a la que pertenece

Nombre de la serie:

Sinopsis:

Limpiar Cancelar Registrar

*Figura 2.2: Crear/Modificar Serie*

## Prototipo de interfaz

Formulario para crear o modificar una temporada. Incluye los siguientes campos:

- Serie a la que pertenece:  (menú desplegable)
- Título de la temporada:
- Número de la temporada:
- Foto de la temporada:
- Es Nueva:
- Forzar publicación:

Botones de acción:

Figura 2.3: Crear/Modificar Temporada

## Prototipo de interfaz

Formulario para crear o modificar un capítulo. Incluye los siguientes campos:

- Capítulo:  (menú desplegable)
- Calendario:  (menú desplegable)
- Serie Pertenece:  (menú desplegable)
- Temporada Pertenece:  (menú desplegable)
- Título del capítulo:
- Número del capítulo:
- Calidad Visual:  (menú desplegable)
- Es Nuevo:
- Protocolo de Transmisión:  (menú desplegable)

Botones de acción:

Figura 2.4: Crear/Modificar Capítulo

Tarea de Ingeniería	
Número Tarea: 1	Número Historia de Usuario: 04
Nombre Tarea: Insertar secciones de series	
Tipo de Tarea: Desarrollo	Puntos Estimados: 0.1
Fecha Inicio:	Fecha Fin:
Programador Responsable: Francisco Izquierdo García	
Descripción: Personalizar formulario de inserción de secciones de series, añadiendo mensajes de validación. Se redefine el método doSave de la clase modelo TseccionSerie para el caso del forzado de publicación de la sección de series.	

Tarea de Ingeniería	
Número Tarea: 2	Número Historia de Usuario: 04
Nombre Tarea: Insertar series	
Tipo de Tarea: Desarrollo	Puntos Estimados: 0.1
Fecha Inicio:	Fecha Fin:
Programador Responsable: Francisco Izquierdo García	
Descripción: Personalizar formulario de inserción de series, añadiendo mensajes de validación. Se redefine el método doSave de la clase modelo Tserie para el caso del forzado de publicación de la serie.	

Tarea de Ingeniería	
Número Tarea: 3	Número Historia de Usuario: 04
Nombre Tarea: Insertar temporadas	
Tipo de Tarea: Desarrollo	Puntos Estimados: 0.1
Fecha Inicio:	Fecha Fin:
Programador Responsable: Francisco Izquierdo García	
Descripción: Personalizar formulario de inserción de temporadas, añadiendo mensajes de validación y se cambia el componente de formulario para carga de imágenes. Se usa librería WebToolkit.js para simular envío de imágenes al servidor vía AJAX haciendo uso de un iframe oculto.	

Tarea de Ingeniería	
Número Tarea: 4	Número Historia de Usuario: 04
Nombre Tarea: Insertar capítulos	

Tipo de Tarea: Desarrollo	Puntos Estimados: 0.1
Fecha Inicio:	Fecha Fin:
Programador Responsable: Francisco Izquierdo García	
Descripción: Crear formulario para insertar capítulos. Se personalizan los mensajes de validación del formulario. Se implementa listas dependientes para la selección de la serie y temporada del capítulo. Se integra con el formulario de calendario de publicación sin periodicidad.	

Tarea de Ingeniería	
Número Tarea: 5	Número Historia de Usuario: 04
Nombre Tarea: Cambiar orden de visualización de las secciones de series publicadas.	
Tipo de Tarea: Desarrollo	Puntos Estimados: 0.1
Fecha Inicio:	Fecha Fin:
Programador Responsable: Francisco Izquierdo García	
Descripción: Se utiliza componente visual de la librería JQuery UI para cambiar el orden de visualización de las secciones de series publicadas.	

Tarea de Ingeniería	
Número Tarea: 6	Número Historia de Usuario: 04
Nombre Tarea: Eliminar temporadas	
Tipo de Tarea: Desarrollo	Puntos Estimados: 0.1
Fecha Inicio:	Fecha Fin:
Programador Responsable: Francisco Izquierdo García	
Descripción: Se redefine el método doDeleteOnCascade de la clase TtemporadaPeer para ganar en rendimiento en la eliminación de temporadas de series y sus capítulos asociados.	

## 2.2.2 Historia de Usuario: Gestionar Carteleras

Historia de Usuario	
Número: 01	Nombre Historia de Usuario: Gestionar Cartelera.
Modificación de Historia de Usuario Número: 3	
Usuario: Javier Méndez Pérez	Iteración Asignada: Iteración 2
Prioridad en Negocio: Muy alta	Puntos Estimados: 2 semanas
Riesgo en Desarrollo: Alto	Puntos Reales: 2 semanas
Descripción: Esta sección garantiza la gestión de las carteleras de juventud rebelde, cartelera de radio ciudad digital y cartelera del canal cultural. Permite además agregar nuevas carteleras con una fecha de	

publicación que contiene una programación de diversos programas que se transmiten a través de los canales.

Observaciones:

1. La cartelera digital de juventud rebelde es descargado de la página del periódico y luego automatizado el proceso de descarga e inserción en la base de datos.
2. Se permitirá crear nuevos canales para las carteleras
3. El sistema debe permitir crear fechas para las carteleras de los programas que se publicarán.
4. Se crearán programas que saldrán en carteleras con una fecha determinada.

### Prototipo de interfaz

Nombre del canal

Limpiar Cancelar Registrar

Figura 2.5: Crear o Modificar Canal

### Prototipo de interfaz

Fecha de cartelera

▼ ▼ ▼ ...

Usar calendario desplegable

Limpiar Cancelar Registrar

Figura 2.6: Crear o Modificar Fecha de Cartelera

## Prototipo de interfaz

Nombre del programa

Seleccione el canal  ▼

Seleccione horario  ▼ :  ▼

Resumen

Imagen

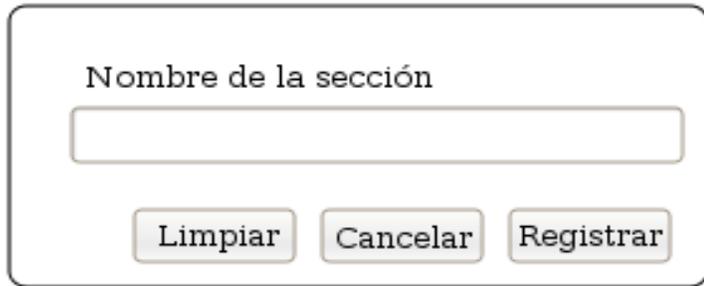
*Figura 2.7: Crear o Modificar Cartelera*

Tarea de Ingeniería	
Número Tarea: 1	Número Historia de Usuario: 01
Nombre Tarea: Insertar fechas de cartelera	
Tipo de Tarea: Desarrollo	Puntos Estimados: 0.1
Fecha Inicio:	Fecha Fin:
Programador Responsable: Javier Méndez Pérez	
Descripción: Se hace uso del plugin para symfony sfFormExtraPlugin para mostrar el componente de selección de fechas. Se personalizan los mensajes de validación del formulario de inserción.	

Tarea de Ingeniería	
Número Tarea: 2	Número Historia de Usuario: 01
Nombre Tarea: Insertar canal	
Tipo de Tarea: Desarrollo	Puntos Estimados: 0.1
Fecha Inicio:	Fecha Fin:
Programador Responsable: Javier Méndez Pérez	
Descripción: Se personalizan los mensajes de validación del formulario de inserción.	

Tarea de Ingeniería	
Número Tarea: 3	Número Historia de Usuario: 01
Nombre Tarea: Insertar cartelera	
Tipo de Tarea: Desarrollo	Puntos Estimados: 0.1
Fecha Inicio:	Fecha Fin:
Programador Responsable: Javier Méndez Pérez	
Descripción: Se redefine el método doSave del formulario TcarteleraForm para guardar la imagen de la cartelera. Se usa librería WebToolkit.js para simular envío de imágenes al servidor vía AJAX haciendo uso de un iframe oculto.	

### 2.2.3 Historia de Usuario: Gestionar Películas

Historia de Usuario	
Número: 02	Nombre Historia de Usuario: Gestionar sección de películas
Modificación de Historia de Usuario Número: 6	
Usuario: Javier Méndez Pérez	Iteración Asignada: Iteración 2
Prioridad en Negocio: Muy alta	Puntos Estimados: 2 semanas
Riesgo en Desarrollo: Alto	Puntos Reales: 2 semanas
Descripción: Esta sección garantiza toda la gestión de las películas, así como su publicación.	
Observaciones: <ol style="list-style-type: none"><li>1. Se integra con la HU Calendario de Publicación</li><li>2. Reutilizará además el componente de autocompletamiento para los campos género y países de la película.</li><li>3. Verificar antes de publicar que en la ruta física este el archivo de media</li><li>4. Sección eliminar   Si tiene activado la eliminación automática de la ficha, siempre dejar las dos últimas películas publicadas</li><li>5. Los conceptos de país y género pueden contener más de un valor asociado, pero uno sólo puede ser el principal.</li><li>6. Los conceptos del director y actor pueden contener más de un valor asociado.</li></ol>	
Nota: Para el campo de actores y directores, tener presente que todos los actores se introducirán en un solo campo de texto, separados por coma (,).	
<b>Prototipo de interfaz</b>	
	
<p><i>Figura 2.8: Crear o Modificar Sección de Películas</i></p>	

## Prototipo de interfaz

Película | Calendario

Título original:

Título al español:

Calidad Visual:   Año:  Duración (min):

Países

Géneros Cinematográficos

Actores:

Directores:

Sinopsis:

Imagen:

Sección pertenece:

Protocolo de Transmisión de la media:

Ruta del archivo en el servidor:

Figura 2.9: Crear o Modificar Película

Tarea de Ingeniería	
Número Tarea: 1	Número Historia de Usuario: 02
Nombre Tarea: Insertar secciones de películas	
Tipo de Tarea: Desarrollo	Puntos Estimados: 0.1
Fecha Inicio:	Fecha Fin:
Programador Responsable: Javier Méndez Pérez	
Descripción: Se personalizan los validadores del formulario de inserción de secciones de películas.	

Tarea de Ingeniería	
Número Tarea: 2	Número Historia de Usuario: 02
Nombre Tarea: Insertar películas	
Tipo de Tarea: Desarrollo	Puntos Estimados: 0.1
Fecha Inicio:	Fecha Fin:
Programador Responsable: Javier Méndez Pérez	
Descripción: Se redefine el método doSave del formulario TpeliculaForm para guardar la imagen de la película y el calendario de publicación asociado. Se implementan nuevos widgets y validadores para la inserción de países y géneros de películas. Se asocia con el calendario de publicación sin periodicidad.	

Tarea de Ingeniería	
Número Tarea: 3	Número Historia de Usuario: 02
Nombre Tarea: Eliminar películas	
Tipo de Tarea: Desarrollo	Puntos Estimados: 0.1
Fecha Inicio:	Fecha Fin:
Programador Responsable: Javier Méndez Pérez	
Descripción: Se redefine el método doDelete de la clase Tpelicula para eliminar la imagen asociada a la película del directorio imágenes.	

## 2.2.4 Historia de Usuario: Gestionar Calendario de Publicación

Historia de Usuario	
Número: 03	Nombre Historia de Usuario: Gestión del calendario de publicación
Modificación de Historia de Usuario Número: 4	
Usuario: Javier Méndez Pérez	Iteración Asignada: Iteración 1
Prioridad en Negocio: Muy alta	Puntos Estimados: 2 semanas
Riesgo en Desarrollo: Alto	Puntos Reales: 2 semanas
Descripción: Esta sección garantiza toda la gestión de la publicación de las medias, específicamente el calendario de publicación.	
Observaciones:	
<ol style="list-style-type: none"> <li>1. Se integra con todas las HU de gestión de medias.</li> <li>2. Luego de finalizar la publicación  Si tiene activado la opción de eliminación de la ficha media, automáticamente se eliminará la ficha de media.</li> <li>3. La opción “Horario de la publicación” solo se hará cuando se determine alguna periodicidad(Por días del mes(Mensualmente),Por días de la semana (Semanalmente) ó Cada x días).</li> <li>4. Cuando se requiere alguna periodicidad se darán los siguientes casos: <ul style="list-style-type: none"> <li>● Si se escoge mensualmente se podrán elegir varios días, los que serán los días del mes que esta media saldrá publicada.</li> <li>● Si se escoge “Por días de la semana” se podrán elegir varios días de la semana a la vez en que se publicará la media.</li> <li>● Si se quisiera publicar una media con una periodicidad diaria, entonces se escogerá la opción Semanalmente, marcando todos los días de la semana.</li> </ul> </li> <li>5. Solo si se elige alguna periodicidad de las anteriores se podrá acceder a la opción de “Horario de la publicación”.</li> <li>6. En la opción de “Horario de la publicación” se debe orientar la hora de inicio y fin de publicación de la media cada vez que se publique.</li> <li>7. La periodicidad solo se realizará si la fecha está en el intervalo de fecha de inicio y fecha de fin.</li> <li>8. Para la fecha de fin solo podrá escoger de las siguientes opciones(1-Tomar fecha por el calendario que incluye la hora o no se selecciona una fecha de fin de la publicación y la media estará publicada permanentemente.</li> <li>9. La opción “No publicar ” si está activada aunque la media esté en un intervalo de publicación no saldrá publicada(Force).</li> <li>10. La opción de “Seleccionar Calendario” debe permitir seleccionar un perfil de calendario creado anteriormente y reutilizarlo para la nueva publicación.</li> </ol>	

## Prototipo de interfaz

Seleccionar calendario  Si selecciona uno de estos calendarios se cargarán los datos a partir del seleccionado

Nombre

No publicar

Comienzo Publicación

Fecha  /  /  ...

Hora  :

Fin Publicación

Fecha  /  /  ...

Hora  :

Luego de finalizar la publicación

Eliminar ficha de media

Periodicidad

Periodicidad

Mensualmente **1**

Semanalmente **2**

Horario de publicación

Inicio  :

Fin  :

Seleccione los días del mes **1**

Adicionar

Quitar

Seleccione los días de la semana **2**

Lunes  Viernes

Martes  Sábado

Miércoles  Domingo

Jueves

Figura 2.10: Calendario Publicación Con Periodicidad

## Prototipo de interfaz

Seleccionar calendario  ▾

Nombre

No publicar

Comienzo Publicación

Fecha  ▾ /  ▾ /  ▾ ...

Hora  ▾ :  ▾

Fin Publicación

Fecha  ▾ /  ▾ /  ▾ ...

Hora  ▾ :  ▾

Luego de finalizar la publicación

Eliminar ficha de media

Si selecciona uno de estos calendarios se cargarán los datos a partir del seleccionado

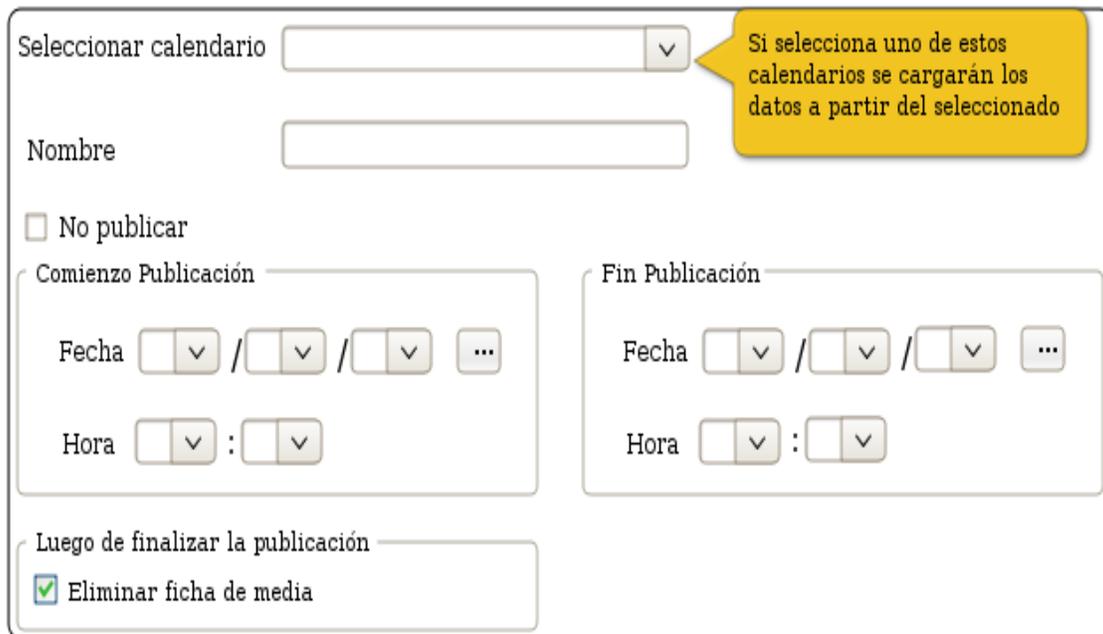


Figura 2.11: Calendario Publicación Sin Periodicidad

Tarea de Ingeniería	
Número Tarea: 1	Número Historia de Usuario: 03
Nombre Tarea: Crear calendario de publicación	
Tipo de Tarea: Desarrollo	Puntos Estimados: 0.1
Fecha Inicio:	Fecha Fin:
Programador Responsable: Javier Méndez Pérez	
Descripción: Se implementa un calendario simple y otro que brinda la posibilidad de escoger determinada periodicidad de la publicación. El calendario se implementa de manera que pueda asociarse a todas las interfaces de publicación de medias.	

### 2.2.5 Historia de Usuario: Gestionar Programas Grabados

Historia de Usuario	
Número: 06	Nombre Historia de Usuario: Gestionar Programas Grabados
Modificación de Historia de Usuario Número: 6	
Usuario: Alexander Martínez Fajardo	Iteración Asignada: Iteración 3
Prioridad en Negocio: Muy alta	Puntos Estimados: 3 semanas
Riesgo en Desarrollo: Alto	Puntos Reales: 2 semanas
Descripción: Esta sección garantiza toda la gestión de la publicación de los programas grabados, realización de encuesta e integración de la encuesta con Programas Grabados.	
Observaciones:	
Glosario:	
Sección: Clasificación por tema del contenido, entiéndase por: Dramatizados, Musicales, Humorísticos, Deportivos, etc.	
Programa: Espacio televisivo generalmente frecuente, que responde a una clasificación en específico, entiéndase por: Colorama, Vale la pena, Entre tu y yo, etc.	
Emisión: Sucesión de programas de un mismo espacio, ejemplo: programa Deja que yo te cuente, las emisiones serían las que se sucederían por días correspondiente a la emisión.	
Descripción de la HU	
Una Sección estará compuesta por Programas y estos a su vez serán agrupados por Emisiones.	
Ejemplo:	
Sección musicales	
<pre>    — Programa Colorama              — Emisión viernes 31/01            — Emisión sábado 4/04   </pre>	

La publicación de los programas deben poder gestionar la cantidad de emisiones que estarán publicadas, pero adicional a esta restricción de cantidad se suma el calendario de publicación que tiene cada emisión, si esa emisión en su calendario se pasa de la fecha, entonces no se mostraría esa emisión dentro del programa al que pertenece. Se debe permitir cambiar el orden de visualización de las secciones publicadas.

Descripción de la sección: Gestionar programas grabados

Sección: {nombre, imagen para icono que la identifique, descripción (puede ser nula), force (activa o desactiva publicación, orden de publicación (puede ser nulo))}

Programas: {nombre, límite visualización de emisiones (5), force (activa o desactiva publicación), descripción (puede ser nula), nuevo (opción que activa una imagen junto al programa que lo identifique como nuevo en publicación)}

Emisión: {«como tal la media», fecha real de emisión en TV (manejarlo con un estándar «dd/mm/aaaa»), descripción, nuevo (opción que activa una imagen junto a la emisión que lo identifique como nuevo en publicación)}

Se debe permitir gestionar Sección, Programas y Emisiones.

Descripción de la sección: Realización de encuesta

Realización de encuesta:

Se muestran en la encuesta todos los programas que están en la base de datos.

Contador de la votación de los usuarios.

Un usuario solo podrá hacer la votación una vez al mes (garantizar autenticación).

Reporte de transcurso de la encuesta

Mostrar reporte de las votaciones de los usuario por programas.

Integración de Encuesta con Programas grabados

Los datos de la encuesta se toman mensuales (de primero de cada mes, hasta el último día de cada mes)

El primero de cada mes pasan a publicación los seis programas ganadores de la encuesta.

Revisar generación automática del tema del calendario de grabación: <http://inter-nos.uci.cu/tv/guia/>

Descripción de la vista general de Programas Grabados:

Se ha definido la creación de un submenu para el trabajo con las categorías sección y emisión. También el listado de las emisiones será a partir de un programa previamente seleccionado, para así limitar el listado a dicho programa. Mediante el listado de secciones se irá desde estas hasta la emisión de un programa, una vez seleccionada la sección, dar clic sobre ella mostrarán los programas relacionados a esta y así mismo con las emisiones de los programas.

Realizar un orden mediante varios criterios de manera que sea mas fácil la búsqueda de una emisión (campos a ordenar: fecha y descripción).

## Prototipo de interfaz

Nombre de la sección:

Icono identificativo:  ...

Breve descripción (Opcional)

Cantidad de caracteres: 0

Forzar Publicación

Figura 2.12: Crear/Modificar Sección de Programas Grabados

## Prototipo de interfaz

Nombre del programa:

Cantidad de emisiones a publicar:

Forzar Publicación  Nuevo programa

Breve descripción: (Opcional)

Cantidad de caracteres: 0

Sección de program  ▼

Figura 2.13: Crear/Modificar Programa Grabado

## Prototipo de interfaz

Emisión | Calendario

Fecha real de emisión en televisión:  
24 / 04 / 2009 ...  Nueva emisión

Breve descripción:

Cantidad de caracteres: 0

Programa de TV

Protocolo de transmisión

Figura 2.14: Crear o Modificar Emisión de Programa Grabado

Tarea de Ingeniería	
Número Tarea: 1	Número Historia de Usuario: 06
Nombre Tarea: Insertar sección de programas grabados	
Tipo de Tarea: Desarrollo	Puntos Estimados: 0.1
Fecha Inicio:	Fecha Fin:
Programador Responsable: Alexander Martínez Fajardo	
Descripción: Se personalizan los validadores de formulario y los mensajes de validación del formulario TseccionPrograbadosForm. Se usa librería WebToolkit.js para simular envío de imágenes al servidor vía AJAX haciendo uso de un iframe oculto.	

Tarea de Ingeniería	
Número Tarea: 2	Número Historia de Usuario: 06
Nombre Tarea: Eliminar sección de programas grabados	
Tipo de Tarea: Desarrollo	Puntos Estimados: 0.1
Fecha Inicio:	Fecha Fin:
Programador Responsable: Alexander Martínez Fajardo	
Descripción: Se redefine el método doDelete de la clase TseccionPrograbados para eliminar la imagen asociada a la sección del directorio imágenes.	

Tarea de Ingeniería	
Número Tarea: 3	Número Historia de Usuario: 06
Nombre Tarea: Insertar programas grabados	
Tipo de Tarea: Desarrollo	Puntos Estimados: 0.1
Fecha Inicio:	Fecha Fin:
Programador Responsable: Alexander Martínez Fajardo	
Descripción: Se personalizan los validadores de formulario y los mensajes de validación del formulario TprograbadoForm.	

Tarea de Ingeniería	
Número Tarea: 4	Número Historia de Usuario: 06
Nombre Tarea: Insertar emisión	
Tipo de Tarea: Desarrollo	Puntos Estimados: 0.1
Fecha Inicio:	Fecha Fin:
Programador Responsable: Alexander Martínez Fajardo	
Descripción: Se personalizan los validadores de formulario y los mensajes de validación del formulario TemisionForm. Se integra con el formulario de calendario de publicación sin periodicidad. Uso del plugin sfFormExtraPlugin para el componente de la fecha de la emisión del programa.	

Tarea de Ingeniería	
Número Tarea: 5	Número Historia de Usuario: 06
Nombre Tarea: Cambiar orden de visualización de las secciones de programas grabados.	
Tipo de Tarea: Desarrollo	Puntos Estimados: 0.1
Fecha Inicio:	Fecha Fin:

Programador Responsable: Alexander Martínez Fajardo
Descripción: Se utiliza componente visual de la librería JQuery UI para cambiar el orden de visualización de las secciones de programas grabados publicados.

## 2.2.6 Historia de Usuario: Gestionar Estaciones de Radio y TV en vivo

Historia de Usuario	
Número: 07	Nombre Historia de Usuario: Gestionar estaciones de radio y TV en vivo
Modificación de Historia de Usuario Número: 4	
Usuario:	Iteración Asignada: 3
Prioridad en Negocio: Muy alta	Puntos Estimados: 3 semanas
Riesgo en Desarrollo: Alto	Puntos Reales: 1 semanas
Descripción: Esta HU garantiza la gestión de las estaciones de radio y TV que se transmiten en Vivo en el portal.	
Observaciones:	
<p>Sección: Adicionar nueva estación de radio o TV en vivo</p> <p>Ha de permitirse la creación de una nueva estación, esta debe comprender: URL, Imagen identificativa y descripción (ambas opcionales), sección a la que pertenece (la sección puede ser: Radio Ciudad Digital   TV en Vivo   Radio Nacional   etc.), forzar la despublicación (esto ha de hacerse para los casos de que no se desee tener publicado en un momento determinado una estación específica). Todas las estaciones tienen asociadas un calendario de publicación.</p> <p>En esta sección, debajo del combobox de las secciones disponibles, deben aparecer las opciones: Adicionar sección (ver sección: Adicionar sección) y Editar sección (ver sección: Actualizar sección)</p> <p>Sección: Adicionar sección de estación de radio o TV en vivo</p> <p>En el caso de que no existiese esa sección, ha de permitirse la creación de una nueva, la cual comprenderá: Nombre de la sección, Imagen identificativa y descripción (ambas opcionales), forzar publicación (esto ha de hacerse para los casos de que se desee tener publicado en un momento determinado una sección específica, aunque no contenga estaciones publicadas. Ha de tenerse en cuenta, que una sección puede estar compuesta por una o mas de una estación y si la sección sale de publicación, no se publicarán la sección y las estaciones).</p> <p>Sección: Actualizar datos en sección de estación de radio o TV en vivo</p> <p>En el caso de que existiese esa sección y se deseara actualizar sus campos, ha de permitirse la creación de una nueva, la cual comprenderá: Nombre de la sección, Imagen identificativa y descripción (ambas opcionales), forzar publicación (esto ha de hacerse para los casos de que se desee tener publicado en un momento determinado una sección específica, aunque no contenga estaciones publicadas. Ha de tenerse en cuenta, que una sección puede estar compuesta por una o mas de una estación y si la sección sale de publicación, no se publicarán la sección y las estaciones). Debe aparecer además un vínculo Eliminar.</p>	

## Prototipo de interfaz

Nombre de la sección:

Imagen identificativa:  ...

Breve descripción (Opcional)

Cantidad de caracteres: 0

Forzar Publicación

Figura 2.15: Crear o Modificar Sección de Estaciones de Radio y TV

## Prototipo de interfaz

Estación | Calendario

Sección a la que pertenece  v

Identificador de la estación

Imagen identificativa  ...

Breve descripción: (Opcional)

Cantidad de caracteres: 0

Protocolo de transmisión  v

Figura 2.16: Crear o Modificar Estaciones de Radio y TV

Tarea de Ingeniería	
Número Tarea: 1	Número Historia de Usuario: 07
Nombre Tarea: Insertar secciones de radio y televisión.	
Tipo de Tarea: Desarrollo	Puntos Estimados: 0.1
Fecha Inicio:	Fecha Fin:
Programador Responsable: Alexander Martínez Fajardo	
Descripción: Se personalizan los validadores de formulario y los mensajes de validación del formulario TseccionEstacionForm. Se usa librería WebToolkit.js para simular envío de imágenes al servidor vía AJAX haciendo uso de un iframe oculto.	

Tarea de Ingeniería	
Número Tarea: 2	Número Historia de Usuario: 07
Nombre Tarea: Eliminar sección de estaciones de radio y televisión	
Tipo de Tarea: Desarrollo	Puntos Estimados: 0.1
Fecha Inicio:	Fecha Fin:
Programador Responsable: Alexander Martínez Fajardo	
Descripción: Se redefine el método doDelete de la clase TseccionEstacion para eliminar la imagen asociada a la sección del directorio imágenes.	

Tarea de Ingeniería	
Número Tarea: 3	Número Historia de Usuario: 07
Nombre Tarea: Insertar estación de radio y televisión	
Tipo de Tarea: Desarrollo	Puntos Estimados: 0.1
Fecha Inicio:	Fecha Fin:
Programador Responsable: Alexander Martínez Fajardo	
Descripción: Se personalizan los validadores de formulario y los mensajes de validación del formulario TestacionForm. Se integra con el formulario de calendario de publicación con periodicidad. Se redefine el método doSave del formulario TestacionForm y el widget correspondiente para guardar la imagen asociada.	

Tarea de Ingeniería	
Número Tarea: 4	Número Historia de Usuario: 07
Nombre Tarea: Eliminar estación de radio y televisión	
Tipo de Tarea: Desarrollo	Puntos Estimados: 0.1

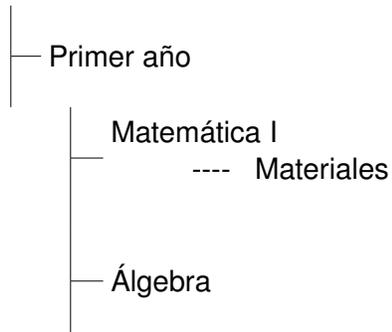
Fecha Inicio:	Fecha Fin:
Programador Responsable: Alexander Martínez Fajardo	
Descripción: Se redefine el método doDelete de la clase TEstacion para eliminar la imagen asociada a la estación del directorio imágenes.	

Tarea de Ingeniería	
Número Tarea: 5	Número Historia de Usuario: 07
Nombre Tarea: Cambiar orden de visualización de las secciones de estaciones publicadas.	
Tipo de Tarea: Desarrollo	Puntos Estimados: 0.1
Fecha Inicio:	Fecha Fin:
Programador Responsable: Alexander Martínez Fajardo	
Descripción: Se utiliza componente visual de la librería JQuery UI para cambiar el orden de visualización de las secciones de estaciones publicadas.	

## 2.2.7 Historia de Usuario: Gestionar Teleclases

Historia de Usuario	
Número: 08	Nombre Historia de Usuario: Gestionar Teleclases
Modificación de Historia de Usuario Número: 3	
Usuario: Javier Méndez Pérez	Iteración Asignada: Iteración 4
Prioridad en Negocio: Muy alta	Puntos Estimados: 1 semana
Riesgo en Desarrollo: Alto	Puntos Reales: 5 días
Descripción: Esta sección garantiza toda la gestión de la publicación de las teleclases docentes.	
Observaciones:	
Glosario:	
Semestre(Período docente): Agrupa los grupos docentes de un periodo, ya sea semestre o un tipo de curso, ejemplo, postgrado, escuela de verano.	
Año: Agrupa un conjunto de asignaturas o tipo de clase, ejemplo, 1er año a 5to año, Postgrado, Escuela de verano, Maestría X.	
Asignatura: Tipo de asignatura.	
Categoría de Material: Tipo de material, ejemplo, Conferencia, Clase Práctica, Laboratorio, Material Complementario.	
Material: Material docente(el archivo de media).	
Descripción de la HU	
Un semestre agrupa un conjunto de años docentes, los cuales a su vez agrupan un conjunto de asignaturas.	
Ejemplo:	

Primer semestre



Descripción de la sección: Gestionar teleclases

Semestre(Periodo docente): {identificador, nombre}

Año docente: {id, nombre, semestre al que pertenece}

Asignatura: {id, nombre, período docente, año al que pertenece}

TipoMaterial: {id, nombrematerial}

Materiales: {id, asignatura a la que pertenece, orden(orden de visualización), título, tipomaterial, sumario(opcional), profesor(opcional)}

Nota: A la hora de insertar el material, el número del orden debe tomar el valor siguiente al último orden insertado. Este campo no se muestra en la interfaz a la hora de insertar el material.

Se debe permitir gestionar Periodo docente, Año docente, Asignatura, TipoMaterial, Materiales.

### Prototipo de interfaz

Período Docente

Limpiar Cancelar Registrar

Figura 2.17: Crear o Modificar Período Docente

**Prototipo de interfaz**

Añó Docente

Limpiar Cancelar Registrar

*Figura 2.18: Crear o Modificar Año Docente*

**Prototipo de interfaz**

Tipo de Material

Limpiar Cancelar Registrar

*Figura 2.19: Crear o Modificar Tipo de Material*

**Prototipo de interfaz**

Asignatura Matemática

Período Docente Semestre 1

Año Docente Primer Año

Limpiar Cancelar Registrar

*Figura 2.20: Crear o Modificar Asignatura*

## Prototipo de interfaz

Material Docente **Calendario**

### Ficha del material

Título del material

Ruta en el servidor

Asignatura a la que pertenece  ▼

Tipo de material  ▼

Sumario (opcional)

Nombre del Profesor (opcional)

### Detalles Técnicos

Ancho Material  Alto Material

Calidad Visual  ▼ Protocolo Transmisión  ▼

Figura 2.21: Crear o Modificar Material

## Prototipo de interfaz

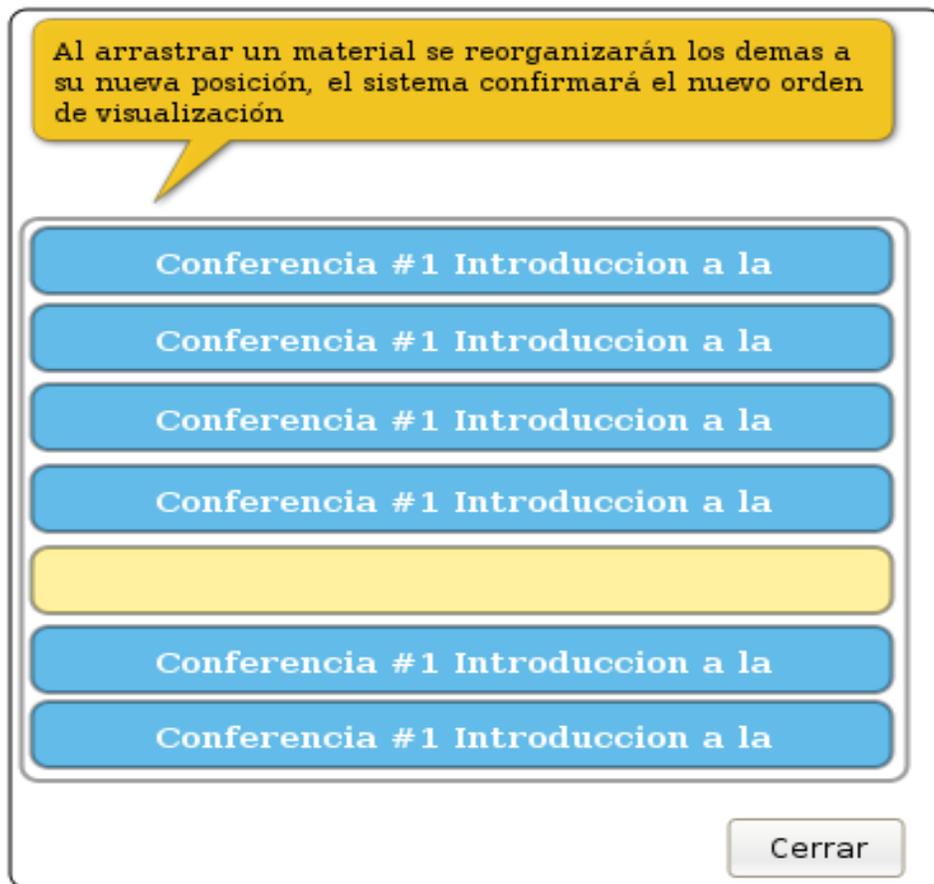


Figura 2.22: Cambiar orden de visualización de materiales docentes

Tarea de Ingeniería	
Número Tarea: 1	Número Historia de Usuario: 08
Nombre Tarea: Insertar período docente	
Tipo de Tarea: Desarrollo	Puntos Estimados: 0.1
Fecha Inicio:	Fecha Fin:
Programador Responsable: Javier Méndez Pérez	
Descripción: Se personalizan los validadores de formulario y los mensajes de validación del formulario.	

Tarea de Ingeniería	
Número Tarea: 2	Número Historia de Usuario: 08
Nombre Tarea: Insertar año docente	
Tipo de Tarea: Desarrollo	Puntos Estimados: 0.1
Fecha Inicio:	Fecha Fin:
Programador Responsable: Javier Méndez Pérez	
Descripción: Se personalizan los validadores de formulario y los mensajes de validación del formulario.	

Tarea de Ingeniería	
Número Tarea: 3	Número Historia de Usuario: 08
Nombre Tarea: Insertar tipos de materiales	
Tipo de Tarea: Desarrollo	Puntos Estimados: 0.1
Fecha Inicio:	Fecha Fin:
Programador Responsable: Javier Méndez Pérez	
Descripción: Se personalizan los validadores de formulario y los mensajes de validación del formulario.	

Tarea de Ingeniería	
Número Tarea: 4	Número Historia de Usuario: 08
Nombre Tarea: Insertar asignaturas	
Tipo de Tarea: Desarrollo	Puntos Estimados: 0.1
Fecha Inicio:	Fecha Fin:
Programador Responsable: Javier Méndez Pérez	
Descripción: Se personalizan los validadores de formulario y los mensajes de validación del formulario. Se redefine el método doSave del formulario TmaterialDocenteForm para asociar un material docente a un protocolo de transmisión.	

Tarea de Ingeniería	
Número Tarea: 5	Número Historia de Usuario: 08
Nombre Tarea: Cambiar orden de visualización de los materiales docentes.	

Tipo de Tarea: Desarrollo	Puntos Estimados: 0.1
Fecha Inicio:	Fecha Fin:
Programador Responsable: Javier Méndez Pérez	
Descripción: Se utiliza componente visual de la librería JQuery UI para cambiar el orden de visualización de los materiales docentes publicados de determinada asignatura.	

## 2.2.8 Historia de Usuario: Gestionar Protocolos de Transmisión

Historia de Usuario	
Número: 09	Nombre Historia de Usuario: Gestionar Protocolos de Transmisión Streaming.
Modificación de Historia de Usuario Número: 3	
Usuario: Alexander y Francisco	Iteración Asignada: Iteración 1
Prioridad en Negocio: Muy alta	Puntos Estimados: 1 semana
Riesgo en Desarrollo: Alto	Puntos Reales: 2 días
Descripción: Esta sección garantiza la gestión de los protocolos de transmisión y nombres de servidores streaming.	
Observaciones:	
Glosario:	
Protocolo de transmisión: Define una parte de la url utilizada como vínculo para visualizar los materiales audiovisuales. Los elementos que lo componen son:	
Nombre del Protocolo: Cadena identificadora del protocolo	
Protocolo de red: Protocolo de red usado por el servidor de media streaming.	
Nombre del servidor de medias: Cadena identificadora con el FQDN(Fully Qualified Domain Name ) del servidor que almacena los archivos de video.	
Puerto: Puerto de escucha del servidor de media streaming.	
Ejemplo de Protocolo de Transmisión:	
mms://ucimedia.uci.cu:80	
http://ucimedia.uci.cu:8080	
Descripción de la sección: Gestionar Protocolos de Transmisión.	
Se debe permitir gestionar los protocolos de transmisión de los materiales.	
Nota: El sistema debe permitir escoger entre los protocolos existentes uno a usar por defecto, que saldrá como primero en el listado de protocolos en el momento de agregar o modificar un material audiovisual. Si no existe ninguno, el que se inserta será establecido por defecto automáticamente.	
Al eliminar un protocolo, si este era el marcado por defecto actualmente, se establece por defecto el primero de los restantes.	

### Prototipo de interfaz

Identificador del Protocolo	<input type="text" value="Protocolo MMS"/>
Protocolo de Red	<input type="text" value="mms"/>
Nombre del Servidor de Streaming	<input type="text" value="ucimedia.uci.cu"/>
Puerto	<input type="text" value="8080"/>
Usar Por Defecto	<input type="checkbox"/>
<input type="button" value="Comprobar Conexión"/>	
<hr/>	
<input type="button" value="Limpiar"/> <input type="button" value="Cancelar"/> <input type="button" value="Registrar"/>	

*Figura 2.23: Crear Protocolo*

### Prototipo de interfaz

Identificador del Protocolo	<input type="text" value="Protocolo MMS"/>
Protocolo de Red	<input type="text" value="mms"/>
Nombre del Servidor de Streaming	<input type="text" value="ucimedia.uci.cu"/>
Puerto	<input type="text"/>
<input type="button" value="Comprobar Conexión"/>	
<hr/>	
<input type="button" value="Limpiar"/> <input type="button" value="Cancelar"/> <input type="button" value="Registrar"/>	

*Figura 2.24: Modificar Protocolo*

Tarea de Ingeniería	
Número Tarea: 1	Número Historia de Usuario: 09
Nombre Tarea: Insertar protocolo de transmisión de las medias.	
Tipo de Tarea: Desarrollo	Puntos Estimados: 0.1
Fecha Inicio:	Fecha Fin:
Programador Responsable: Francisco Izquierdo García	
Descripción: Se personalizan los validadores del formulario TprotocoloForm. Se redefine el método doSave de la clase Tprotocolo para marcar por defecto el primer protocolo insertado. Se redefine el método doDelete de la clase Tprotocolo para el caso que se elimine el protocolo marcado por defecto.	

### 2.2.9 Historia de Usuario: Gestionar Usuarios del Sistema

Historia de Usuario	
Número: 10	Nombre Historia de Usuario: Gestionar Usuarios del Sistema
Modificación de Historia de Usuario Número: 2	
Usuario: Javier Méndez Pérez	Iteración Asignada: 5
Prioridad en Negocio: Muy alta	Puntos Estimados: 2 semanas
Riesgo en Desarrollo: Alto	Puntos Reales: 9 días
Descripción: Garantiza la gestión de los usuarios y grupos de usuarios del sistema, organizados por roles.	
Observaciones: Debe permitir insertar usuarios al sistema tanto del dominio UCI.CU como usuarios locales. Debe permitir agrupar usuarios por roles(grupos de usuarios), asignándole determinados permisos a cada rol, de manera que cada usuario esté limitado a acceder a determinadas funcionalidades de la aplicación según los permisos asignados al rol que pertenece.	
Descripción de la sección: Gestionar usuarios del sistema usuario: {usuario del dominio} usuarios locales {nombre del usuario, primer apellido, segundo apellido y contraseña} grupos de usuario {nombre del grupo de usuario, descripción del grupo y listado de permisos asignados} permiso {nombre del permiso, descripción del permiso, código del permiso}	
Se debe permitir gestionar usuarios, grupos. Se debe permitir modificar el nombre o descripción de un permiso ya establecido.	

### Prototipo de interfaz

Nombre del permiso

Descripción

---

*Figura 2.25: Modificar permiso*

### Prototipo de interfaz

Nombre del grupo

Permisos

- Ver Teleclase
- Crear/Editar/Eliminar Teleclase
- Ver Series
- Crear/Editar/Eliminar Series
- Ver Películas

Descripción

---

*Figura 2.26: Crear o Modificar Grupo*

## Prototipo de interfaz

Usuario	<input type="text"/>
	<input type="checkbox"/> No usuario del dominio
Nombre	<input type="text"/>
Primer Apellido	<input type="text"/>
Segundo Apellido	<input type="text"/>
Contraseña	<input type="text"/>
Elija el(los) grupo(s)	
	<input type="checkbox"/> Publicadores de Series
	<input type="checkbox"/> Publicadores de Películas
<hr/>	
	<input type="button" value="Limpiar"/>
	<input type="button" value="Cancelar"/>
	<input type="button" value="Registrar"/>

*Figura 2.27: Crear o Modificar Usuario*

Tarea de Ingeniería	
Número Tarea: 1	Número Historia de Usuario: 10
Nombre Tarea: Insertar usuarios del sistema	
Tipo de Tarea: Desarrollo	Puntos Estimados: 0.1
Fecha Inicio:	Fecha Fin:
Programador Responsable: Javier Méndez Pérez	
Descripción: Se personalizan los validadores del formulario TusuarioForm y los mensajes de validación.	

Tarea de Ingeniería	
Número Tarea: 2	Número Historia de Usuario: 10
Nombre Tarea: Insertar grupos de usuarios (roles) del sistema.	
Tipo de Tarea: Desarrollo	Puntos Estimados: 0.1
Fecha Inicio:	Fecha Fin:
Programador Responsable: Javier Méndez Pérez	
Descripción: Se personalizan los validadores del formulario. Se redefinen los widget que listan los permisos del sistema.	

## 2.2.10 Historia de Usuario: Servicios Web

Historia de Usuario	
Número: 11	Nombre Historia de Usuario: Servicios Web
Modificación de Historia de Usuario Número: 0	
Usuario: Alexander y Francisco	Iteración Asignada: 5
Prioridad en Negocio: Baja	Puntos Estimados: 1 semanas
Riesgo en Desarrollo: Medio	Puntos Reales: 5 días
Descripción: Garantiza la publicación de la información referente a los materiales audiovisuales a través de servicios web.	
Observaciones: Los servicios web no tendrán restricción para su consumo, de manera que estarán disponibles para toda la comunidad universitaria.  La información que brindarán los servicios web, se mostrará de la misma forma en que se visualiza actualmente en el portal Inter-Nos, a excepción de la información referente a las teleclases; la misma debe brindarse de acuerdo a la nueva organización planteada en la HU Gestionar Teleclases.	

Número Tarea: 1	Número Historia de Usuario: 11
Nombre Tarea: Instalación y configuración del plugin ckWebServicePlugin. Diseño de tipos de datos complejos que representan la información de los materiales a publicar. Implementación de las clases y métodos a publicar.	
Tipo de Tarea: Desarrollo	Puntos Estimados: 5
Fecha Inicio:	Fecha Fin:
Programador Responsable: Alexander y Francisco	
Descripción: Se instala y configura apropiadamente el plugin ckWebServicePlugin para exponer los métodos de la capa de Controlador designados como servicios web. Se diseñan e implementan los tipos de datos complejos a usar en el mensaje SOAP. Implementación de los métodos que se expondrán como servicios web.	

### 2.3 Plan de Releases

De acuerdo a las funcionalidades descritas en las historias de usuarios y en concordancia con la prioridad asignada para su realización, se planificaron 5 iteraciones que recogen el desarrollo de la solución propuesta.

Release	Descripción de la iteración	Orden de la HU a implementar	Duración total
1	Darle cumplimiento a las HU que tienen la máxima prioridad	03, 04, 09	6 semanas
2	Darle cumplimiento a las HU que tienen prioridad muy alta e integrar los componentes desarrollados en la iteración anterior.	01, 02	4 semanas
3	Desarrollar las HU de prioridad alta e integrar los componentes desarrollados en iteraciones anterior	06, 07	6 semanas
4	Darle cumplimiento los requisitos funcionales correspondientes a la gestión de teleclases . Integrar los componentes desarrollados a la solución y diseñar e implementar los servicios web que brindará la solución propuesta.	8	1 semana
5	Desarrollar los componentes relacionados con la seguridad en la aplicación y el control de acceso	10,11	3 semana

## **2.4 Conclusiones parciales**

Hasta aquí se han se identifican en mutuo acuerdo con el cliente un conjunto de requisitos que el sistema debe ser capaz de brindar, desglosados en Historias de Usuario que cada desarrollador debe cumplir a través de una serie de actividades del ciclo de vida del producto o Tareas de Ingeniería; y se establecen los ciclos iterativos en los que serán desarrolladas o mejoradas las funcionalidades de la solución propuesta.

## Capítulo 3 Desarrollo del sistema propuesto

Este capítulo aborda el diseño a alto nivel de la solución propuesta, describiendo el patrón arquitectónico Modelo – Vista – Controlador, en la variante implementada por el framework Symfony; así como patrones de diseño empleados y los mecanismos de seguridad que provee el framework. También se describe la estructura modular del sistema propuesto a través de diagramas de componentes, las interfaces de comunicación externa o servicios web que brindará el mismo; así como la estrategia utilizada para el diseño del modelo de datos y el estándar de código usado en la implementación.

### 3.1 Patrones de diseño y arquitectura del framework Symfony

#### ¿Qué es un patrón de diseño?

“Cada patrón, describe un problema que ocurre una y otra vez en nuestro entorno, y describe el núcleo de la solución al problema, de manera que puede utilizarse un millón de veces, sin hacer lo mismo dos veces”[39]. Los patrones de diseño resuelven problemas comunes a partir de soluciones probadas mediante un diseño orientado a objeto facilitando la reutilización de clases ya existentes. Dentro de los elementos más importantes que componen un patrón, se destacan el nombre del patrón, el problema que resuelve y la solución al mismo. Existen varias clasificaciones para los patrones, aunque las más destacadas son de acuerdo al propósito del patrón: creacionales, estructurales y de comportamiento, y la otra clasificación es de acuerdo al alcance del mismo: a nivel de clases u objetos[40].

#### 3.1.1 Patrones de diseño implementados por Symfony

El framework Symfony implementa una serie de patrones de diseño que hacen su arquitectura sea suficientemente robusta y a la vez flexible como para adaptarse a los casos más complejos. A continuación se especifican algunos patrones y ejemplos de uso dentro de la solución propuesta:

- Singleton (Única instancia): Es un patrón creacional que garantiza la existencia de una única instancia para una clase y la creación de un mecanismo de acceso global a dicha instancia. A través de la clase `SfContext::getInstance()`, se obtiene una referencia a todos los objetos del núcleo del

framework por lo que es posible por ejemplo acceder desde las clases de formularios a métodos definidos en la clase que maneja las peticiones.

- **Front Controller (Controlador Frontal):** Es un patrón de diseño web usado como único punto de entrada a la aplicación. Realiza tareas comunes a todos los controladores, ejemplo, manejo de la seguridad, de las peticiones de los usuarios, carga de la configuración de la aplicación y delega la responsabilidad de responder a las peticiones al módulo específico que tiene la acción enviada en la petición.
- **Decorator (Envoltorio):** Normalmente en las aplicaciones web existen contenidos que son comunes a todas las páginas que conforman la aplicación. Symfony haciendo uso de este patrón de diseño estructural delimita el código común en todas las páginas (definido en un archivo global para todas las vistas denominado layout) del código HTML generado como respuesta a una petición determinada (denominado plantilla). El contenido de la plantilla se integra en el layout, por lo que se puede afirmar que el layout decora la plantilla.
- **Experto:** es un patrón de asignación de responsabilidades (GRASP, del inglés General Responsibility Assignment Software Patterns) que delega las responsabilidades a quién contiene la información necesaria para cumplirlas. Tal es el caso de las clases que genera la librería Propel con todas las operaciones o métodos útiles que consolidan un desarrollo ágil y una abstracción del sistema gestor de bases de datos empleado.

### **3.1.2 Implementación del patrón Modelo Vista Controlador de Symfony**

Symfony está basado en un patrón clásico del diseño web, conocido como arquitectura Modelo Vista Controlador (MVC), que separa la lógica de negocio (modelo) y la presentación (vista), consiguiendo aplicaciones más sencillas de mantener[41]. El patrón MVC fue introducido en los años 70 por desarrolladores de la plataforma SmallTalk y marcó un paso de avance en la arquitectura de las aplicaciones web. El patrón considera tres roles:[42]

- El modelo, representa información relacionada con el dominio de la aplicación.
- La vista, es la representación del modelo en interfaces de usuarios. Solo se encarga de mostrar la información, cualquier cambio es manipulado por el controlador.

- El controlador, interpreta las acciones del teclado o el ratón informando al modelo y/o la vista para que se actualicen según resulte apropiado.

La separación de la vista y el modelo trae ventajas; es posible tener diferentes representaciones de la misma información, haciendo uso del mismo código dentro del modelo. Es posible además programar el código del modelo, abstrayéndose de la representación visual que se le dará a la información.

“Symfony toma lo mejor de la arquitectura Modelo Vista Controlador y la implementa de forma que el desarrollo de aplicaciones sea rápido y sencillo”[43]. La capa referente al controlador dentro de la arquitectura MVC, Symfony lo divide en un controlador frontal y acciones, ubicando el controlador frontal en el directorio web del proyecto. La capa del modelo está dividida en dos capas, una para la abstracción del gestor de bases de datos y la otra es la capa de acceso a los datos, con lo que logra aplicaciones independientes a los sistemas gestores de base de datos.

Una gran ventaja que tiene Symfony con respecto a muchos otros frameworks de desarrollo radica en el empleo de tareas automatizadas a través de la interfaz de líneas de comandos de Symfony y el uso de la librería Propel para el mapeo de objetos relacionales (ORM), encargada de la generación automatizada de las clases necesarias para el acceso a los datos, a partir del esquema de la base de datos escrito en XML o YAML. Por otra parte la capa de vista, está separada en tres partes: los elementos comunes a todas las páginas de la aplicación, denominada layout; las plantillas, encargadas de visualizar las variables definidas en el controlador; y la lógica de las vistas, definida a través de los slots y componentes (fragmentos de vistas que contienen lógica de aplicación y pueden ser reutilizables).

### **3.1.3 Estructura del framework Symfony**

Symfony considera un proyecto como "un conjunto de servicios y operaciones disponibles bajo un determinado nombre de dominio y que comparten el mismo modelo de objetos". Un proyecto desarrollado con Symfony se compone de una o más aplicaciones, y estas a su vez están organizadas en módulos. Los módulos encapsulan código que está estrechamente relacionado, y representan operaciones que se realizan en los mismos. Symfony implementa una estructura jerárquica de directorios que permite realizar configuraciones a distintos niveles: proyecto, aplicación o módulo; con el que se logra un nivel de configuración muy preciso. El único punto de acceso a las

aplicaciones del proyecto, es el directorio web, en el que se almacenan las imágenes, hojas de estilos y ficheros JavaScript compartidos por las mismas.

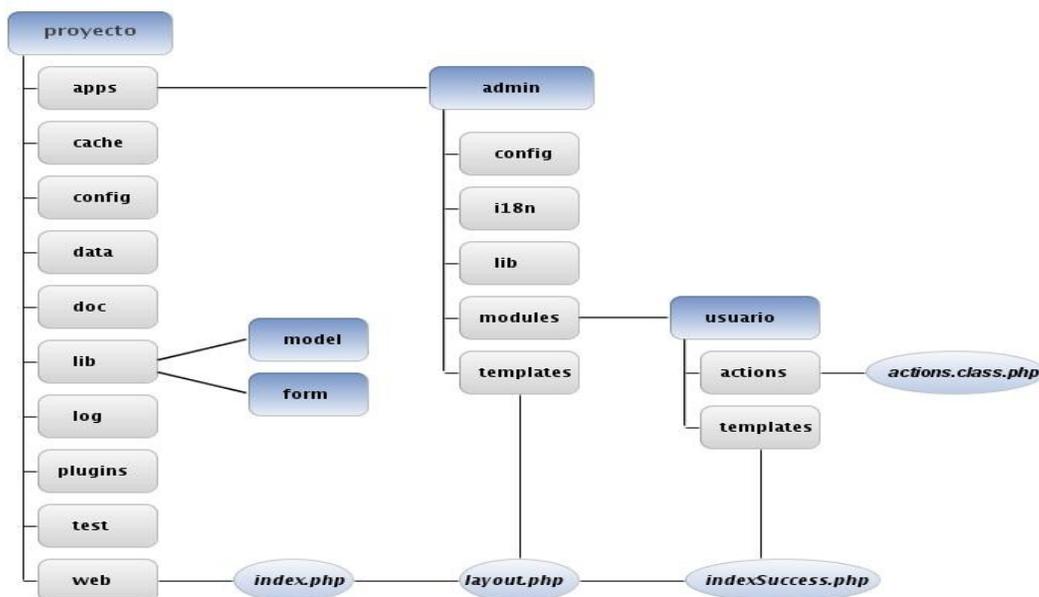


Figura 3.1: Estructura de directorios de un proyecto Symfony

### 3.2 Seguridad en las aplicaciones desarrolladas con Symfony

En el desarrollo de aplicaciones web, uno de los aspectos más importantes, es la garantía de la confidencialidad, integridad y disponibilidad de la información. Las aplicaciones web se ven afectadas en este sentido por una serie de vulnerabilidades del propio código fuente, muchas veces por descuido o malas prácticas de programación. En este sentido Symfony implementa una serie de mecanismos para garantizar aplicaciones seguras. Tal es el caso de la protección de los datos contra ataques CSRF, Cross Site Scripting (XSS) y SQL Injection.

Los ataques de tipo XSS, aprovechan vulnerabilidades en el sistema de validación de los componentes HTML incrustados en una página web, provocando muchas veces la denegación de servicios y pérdida de información. Los ataques de tipo CSRF (del inglés, Cross-site request forgery), aprovecha la confianza que tienen los sitios web en los usuarios, para ejecutar códigos maliciosos. Es conocido también como enlace hostil, ataque de un clic o cabalgamiento de sesión. Los sistemas informáticos se ven afectados además por ataques de inyección SQL, que explotan vulnerabilidades

en los códigos de acceso a datos con el objetivo de obtener claves de acceso de los usuarios para suplantar la identidad de los mismos o destruir bases de datos completas. Para evitar estos ataques, Symfony implementa una serie de mecanismos de seguridad. Con el fin de contrarrestar los ataques de tipo XSS, se establecen estrategias de escape para las variables disponibles en las vistas, provocando se apliquen una serie de funciones de PHP que evitan la ejecución de código Javascript malicioso. En el caso de los ataques de tipo CSRF, se establece la directiva `csrf_secret`, a la misma se le asigna una palabra secreta; internamente el framework genera un hash a partir del token establecido, el cual puede ser comprobado en cada petición, incluyéndolo como valor en un widget de formulario o en los helpers para vínculos. Es en el fichero `settings.yml` de la aplicación donde se configuran estas directivas.

```
.settings:
  # Form security secret (CSRF protection)
  csrf_secret:      PaLaBraSeCreta| # Uni

  # Output escaping settings
  escaping_strategy:    on          # De
  escaping_method:     ESC_SPECIALCHARS ;
```

*Figura 3.2: Protección contra ataques XSS y CSRF*

Uno de los mecanismos de seguridad establecidos por Symfony es aplicar una serie de filtros antes de la ejecución de una acción determinada. Uno de esos filtros es el de restricción de acceso, mediante el cual se chequea que las acciones que son seguras sean ejecutadas sólo por usuarios autenticados y con privilegios sobre las mismas. Este mecanismo es fácil de configurar y muy preciso a la hora de aplicarse tanto a nivel de aplicación como de módulo, aprovechando las ventajas de la configuración en cascada que implementa el framework. A continuación un ejemplo de un archivo de configuración de seguridad `security.yml` para un módulo de una aplicación Symfony.

```

ver:
  is_secure: off      # Todos Los usuarios pueden ejecutar la acción "ver"

modificar:
  is_secure: on      # La acción "modificar" es sólo para usuarios autenticados

borrar:
  is_secure: on      # Sólo para usuarios autenticados
  credentials: admin # Con credencial "admin"

all:
  is_secure: off     # off es el valor por defecto

```

*Figura 3.3: Configuración de seguridad para las acciones en security.yml*

En caso de que un usuario que intente ejecutar una acción y no esté autenticado en el sistema, automáticamente será redirigido hacia una acción establecida por defecto en el fichero settings.yml de la aplicación. En este fichero las directivas login\_action y login\_module establecen la acción y módulo respectivamente que serán ejecutados cuando ocurren las condiciones anteriormente explicadas. Además es posible establecer un módulo y una acción por defecto que será ejecutada cuando un usuario autenticado intenta realizar una acción marcada como segura pero no tiene los privilegios necesarios para ejecutarla. Las directivas que establecen qué acción será ejecutada son secure\_module y secure\_action.

```

all:
  .actions:
    login_module: default
    login_action: login

    secure_module: default
    secure_action: secure

```

*Figura 3.4: Configuraciones de seguridad en el fichero settings.yml*

### **3.3 Arquitectura de la solución propuesta**

Para darle cumplimiento al problema planteado se creó un proyecto Symfony que consta de dos aplicaciones:

- **internos:** aplicación con la que interactúan los usuarios del sistema, dedicada a la gestión de los materiales publicados en el portal Inter-Nos. Cuenta con un total de 18 módulos, a los cuales se les establecen los privilegios de acceso que los usuarios deberán tener para operar sobre los mismos.
- **webservices:** aplicación dedicada a brindar los servicios web al portal Inter-Nos. Posee 6 módulos que publican a través de los servicios web la información referente a series, películas, programas grabados, radio y televisión, carteleras y teleclases.

#### **3.3.1 Diagrama de componentes**

En un sistema es tan importante la modelación de los aspectos lógicos y de diseño como aquellos aspectos de implementación. “La vista de implementación muestra el empaquetado físico de las partes reutilizables del sistema en unidades sustituibles, llamadas componentes. Una vista de implementación muestra la implementación de los elementos del diseño (tales como clases) mediante componentes, así como sus interfaces y dependencias entre componentes. Los componentes son las piezas reutilizables de alto nivel a partir de las cuales se pueden construir los sistemas” [44]. A continuación se describen los componentes de la solución propuesta, así como su interacción con otros componentes.

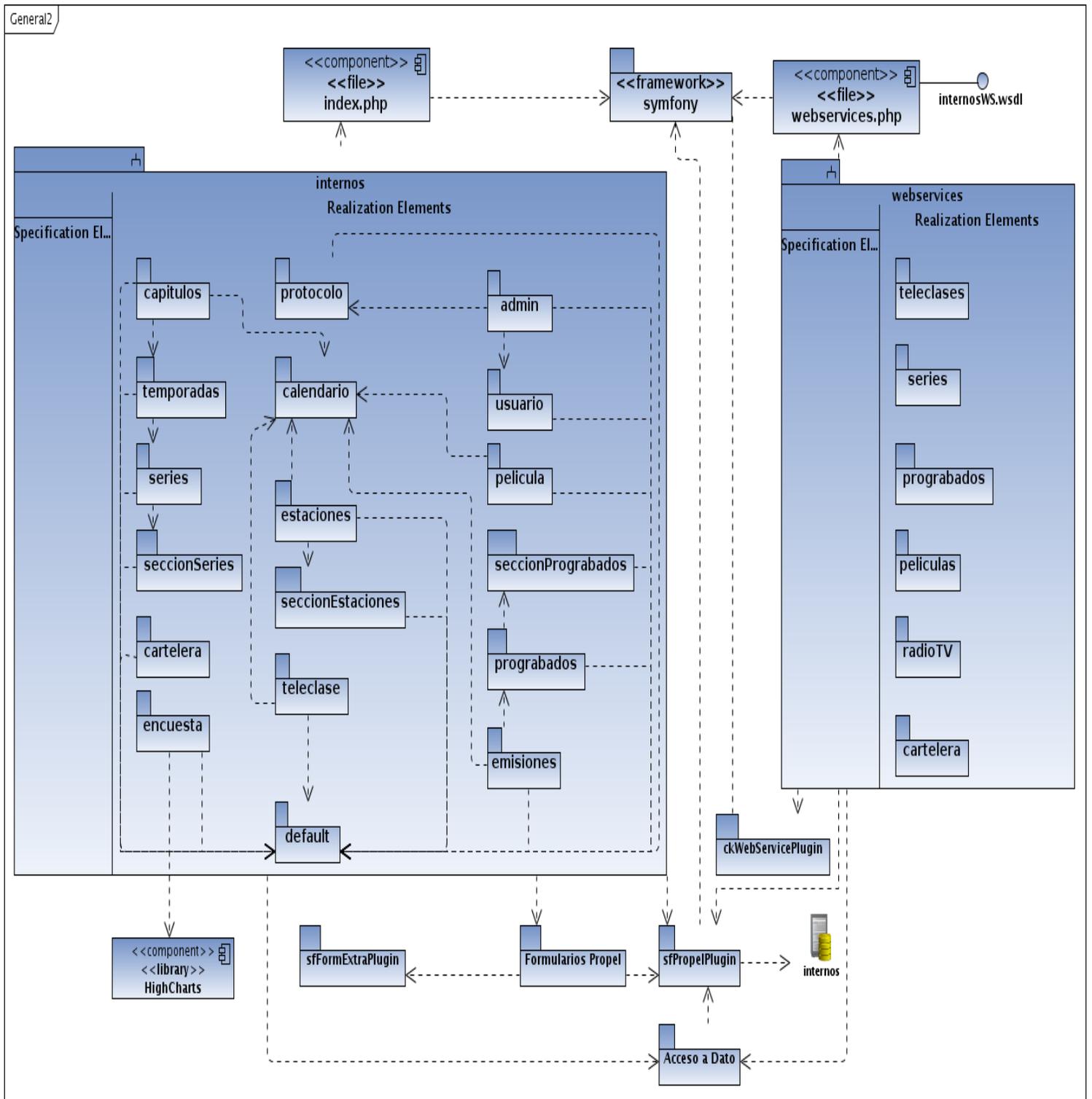


Figura 3.5: Diagrama de Componentes de la solución propuesta

## Descripción de los módulos y sus funcionalidades

### Aplicación internos

**Módulo protocolo:** encargado de la gestión de los diferentes protocolos de transmisión de los materiales audiovisuales en la red.

- Listar protocolos de transmisión.
- Crear protocolos de transmisión.
- Editar protocolos de transmisión existentes.
- Eliminar protocolos de transmisión.
- Establecer protocolos de transmisión como predeterminado.
- Mostrar detalles de un protocolo de transmisión.

**Módulo usuario:** encargado de la gestión de los usuarios y grupos del sistema.

- Listar usuarios y/o grupos del sistema.
- Crear usuarios y/o grupos del sistema.
- Modificar usuarios y/o grupos del sistema.
- Eliminar usuarios y/o grupos del sistema.
- Mostrar detalles de usuarios y/o grupos del sistema.
- Listar permisos del sistema
- Mostrar detalles de los permisos del sistema.
- Editar los permisos del sistema (sólo la descripción y nombre del permiso).
- Autenticar los usuarios del sistema.

**Módulo admin:** es el módulo desde el cual se accede a la gestión de los usuarios y grupos del sistema y los protocolos de transmisión. Constituye la interfaz de acceso a dichos módulos.

**Módulo default:** es el módulo de acceso a la aplicación y contiene vistas que se reutilizan en los demás módulos, así como las plantillas personalizadas de error 404, aplicación deshabilitada, y la plantilla que informa al usuario que no tiene los privilegios necesarios para realizar determinada

acción.

**Módulo cartelera:** es el módulo desde el que se gestionan las carteleras de los canales de la Televisión Cubana y los canales universitarios.

- Insertar, modificar, listar y eliminar fechas de carteleras.
- Insertar, modificar, listar y eliminar canales.
- Ver carteleras de un canal.
- Insertar, modificar, listar y eliminar carteleras.
- Mostrar detalles de una cartelera.

**Módulo calendario:** es utilizado por otros módulos del sistema para establecer los calendarios de publicación de los materiales audiovisuales. Contiene plantillas de los diferentes tipos de calendarios (con periodicidad y sin periodicidad).

- Copiar calendario creado.
- Eliminar calendarios.

**Módulo seccionSeries:** encargado de la gestión de las secciones de series publicadas en Inter-Nos.

- Insertar, modificar, listar y eliminar secciones de series.
- Forzar publicación de secciones de series.
- Visualizar sólo secciones de series publicadas.
- Cambiar orden de visualización de las secciones de series publicadas.

**Módulo series:** encargado de la gestión de las series pertenecientes a una sección de series determinada.

- Insertar, modificar, listar y eliminar series.
- Forzar publicación de series.
- Mostrar detalles de una serie.

**Módulo temporadas:** módulo encargado de la gestión de las temporadas de una serie determinada.

- Insertar, modificar, listar y eliminar temporadas de series.

- Forzar publicación de temporadas.
- Mostrar detalles de una temporada determinada.

**Módulo capítulos:** módulo encargado de la gestión de los capítulos de las series. Hace uso del módulo calendario para establecer fechas de publicación de los capítulos.

- Insertar, modificar, listar y eliminar capítulos de series.
- Mostrar detalles de un capítulo.

**Módulo película:** encargado de la gestión de las secciones de películas y las películas que se publican en el portal Inter-Nos. Se asocia con el módulo calendario para establecer fechas de publicación de las películas.

- Insertar, modificar, listar y eliminar secciones de películas.
- Insertar, modificar, listar y eliminar películas de una sección de películas.
- Mostrar detalles de una película.

**Módulo seccionEstaciones:** es el encargado de gestionar las secciones de estaciones de radio y televisión que se publican en el portal Inter-Nos.

- Insertar, modificar, listar y eliminar secciones de estaciones de radio o televisión.
- Visualizar sólo las secciones de estaciones de radio o televisión publicadas.
- Cambiar orden de visualización de las secciones de estaciones de radio o televisión.
- Mostrar detalles de una sección de estaciones de radio o televisión.
- Forzar publicación de secciones de estaciones de radio o televisión.
- Listar las estaciones pertenecientes a una sección de estaciones de radio o televisión.

**Módulo estaciones:** módulo desde el que se gestionan las estaciones de radio y televisión que son publicadas en el portal Inter-Nos. Se asocia con el módulo calendario para establecer fechas de publicación de las estaciones.

- Insertar, modificar, listar y eliminar estaciones de radio o televisión.
- Mostrar detalles de una estación de radio o televisión.

**Módulo seccionPrograbados:** encargado de la gestión de las secciones de programas grabados.

- Insertar, modificar, listar y eliminar secciones de programas grabados.
- Visualizar sólo las secciones de programas grabados publicadas.
- Cambiar orden de visualización de las secciones de programas grabados publicadas.
- Mostrar detalles de una sección de programas grabados.
- Forzar publicación de secciones de programas grabados.
- Listar los programas grabados pertenecientes a una sección de programas grabados.

**Módulo prograbados:** encargado de la gestión de los programas grabados que se publican en el portal Inter-Nos.

- Insertar, modificar, listar y eliminar programas grabados.
- Mostrar detalles de los programas grabados.
- Forzar publicación de los programas grabados.
- Listar las emisiones pertenecientes a un programa grabado.

**Módulo emisiones:** encargado de gestionar las emisiones de los programas grabados. Se asocia con el módulo calendario para establecer fechas de publicación de las emisiones de los programas grabados.

- Insertar, modificar, listar y eliminar las emisiones de los programas grabados.
- Mostrar detalles de las emisiones de los programas grabados.

**Módulo encuesta:** encargado de la gestión de las encuestas de los programas grabados que se publicarán en el portal Inter-Nos.

- Insertar, listar y eliminar las encuestas de los programas grabados.
- Mostrar los resultados de las encuestas de los programas grabados hasta la fecha actual.
- Mostrar un numero determinado de programas ganadores de una encuesta determinada.

**Módulo teleclase:** encargado de la gestión de las teleclases que se publican en el portal Inter-Nos.

- Insertar, modificar, listar y eliminar períodos docentes.

- Insertar, modificar, listar y eliminar años docentes.
- Insertar, modificar, listar y eliminar tipos de materiales docentes (conferencias, clases prácticas, etc).
- Insertar, modificar, listar y eliminar asignaturas.
- Insertar, modificar, listar y eliminar materiales docentes.
- Mostrar detalles de los materiales docentes.
- Cambiar el orden de visualización de los materiales docentes.

### **Aplicación webservices**

**Módulo series:** brinda la información a través de servicios web de las secciones de series, series, temporadas y capítulos en fecha de publicación.

**Módulo películas:** brinda la información a través de servicios web de las secciones de películas y películas en fecha de publicación.

**Módulo grabados:** brinda la información a través de servicios web de las secciones de programas grabados, así como los programas grabados y sus emisiones en fecha de publicación. Además publica la encuesta del mes en curso, recoge los votos de los usuarios y brinda los resultados de la misma hasta la fecha actual. El módulo también publica los programas ganadores de la encuesta del mes anterior.

**Módulo radioTV:** brinda la información a través de servicios web de las secciones de estaciones de radio y televisión y estaciones de radio y televisión en fecha de publicación.

**Módulo cartelera:** encargado de brindar la información a través de los servicios web de las carteleras de los canales de la Televisión Cubana y los canales universitarios.

**Módulo teleclases:** encargado de brindar la información a través de los servicios web de los períodos docentes, años docentes, asignaturas y materiales docentes publicados en el portal Inter-Nos.

### **3.3.2 Esquema de la base de datos utilizada en la solución propuesta**

A continuación se presenta el modelo de datos utilizado por la aplicación y la estrategia seguida para

la obtención del mismo. Para la construcción del diagrama entidad-relación fue utilizada la herramienta CASE Visual Paradigm for UML versión 6.4. Luego se realizaron las transformaciones necesarias para obtener el modelo relacional, y por último se normalizó parcialmente (debido a requerimientos del cliente) el esquema obtenido hasta la tercera forma normal. Posteriormente se escribió en el archivo *schema.xml* el esquema final que utiliza la librería Propel para la generación automática de las clases modelo y formularios y la inserción de la base de datos en el Sistema Gestor de Bases de Datos MySQL (**Ver Anexo II**).

### **3.4 Descripción de las interfaces de comunicación externas (Servicios Web)**

Con el fin de hacer disponible la información gestionada a través de la aplicación se implementaron un serie de servicios web que serán consumidos por el portal Inter-Nos y además por el resto de la comunidad universitaria desde cualquier aplicación cliente.

#### **3.4.1 Complemento ckWebServicePlugin**

Mediante el uso del plugin ckWebServicePlugin se pueden exponer los métodos de la capa del controlador como servicios web, aprovechando las funcionalidades del framework Symfony referentes al manejo de variables de las peticiones y la capa de acceso a dato. El plugin además genera un documento WSDL en el que incluye todas las funcionalidades disponibles como servicios web dentro de la aplicación.

Para el uso del mismo, se realizan una serie de configuraciones a nivel de aplicación en los ficheros *factories.yml*, *settings.yml* y *filters.yml*. Se diseñan e implementan las clases que representan los tipos de datos complejos a usar en los mensajes SOAP y por último los métodos de la capa del controlador que se deseen incluir como servicios web. Para lograr esto se marcan con un bloque de documentación especial que los identifica; en el mismo se incluye el nombre que tendrá el método en el documento WSDL, los parámetros en caso que el método lo requiera y el tipo de dato que retorna.

Una vez realizado el proceso descrito anteriormente se procede a generar el documento WSDL a través de una tarea de línea de comando que se incluye con la instalación del plugin. La tarea requiere de una serie de parámetros para su ejecución, los cuales incluyen, el nombre de la aplicación Symfony, el nombre de la interfaz que expondrá los métodos y la dirección url del servicio web. Para la solución propuesta la tarea queda de la siguiente manera:

```
./symfony generate:wSDL webservices internosWS http://internosAdmin.uci.cu/webService
```

Figura 3.6: Tarea para generar documento WSDL

### 3.4.2 Tipos de datos complejos usados

A continuación se describen los tipos de datos complejos empleados como respuestas en los mensajes SOAP.

Clase: Canal		Módulo: cartelera	
Descripción: Clase encargada de representar información de canales de televisión.			
Atributo	Tipo	Descripción	
idCanal	int	Identificador del canal	
nombre	string	Nombre del canal	

Clase: Cartelera		Módulo: cartelera	
Descripción: Clase encargada de representar información de carteleras de programas de televisión			
Atributo	Tipo	Descripción	
idCartelera	int	Identificador de la cartelera	
horaCartelera	string	Hora de la cartelera	
nombrePrograma	string	Nombre del programa de televisión	
resumenPrograma	string	Resumen del programa en cartelera	
imagenCartelera	string	Imagen identifica al programa en cartelera	

Clase: FechaCartelera		Módulo: cartelera	
Descripción: Clase encargada de representar información de la fecha de las carteleras de los programas. Incluye una colección de carteleras para esa fecha.			
Atributo	Tipo	Descripción	
idFecha	int	Identificador de la fecha	
fecha	string	Fecha de la cartelera (formato año-mes-día)	
carteleras	Cartelera[]	Colección de carteleras para la fecha	

Clase: SeccionPelículas Módulo: películas

Descripción: Clase encargada de representar información una sección de películas

Atributo	Tipo	Descripción
idSeccionPelículas	int	Identificador de la sección de películas
tituloSeccionPelículas	string	Título de la sección de películas

Clase: Película Módulo: películas

Descripción: Clase encargada de representar información de una película

Atributo	Tipo	Descripción
idPelícula	int	Identificador de la película
titulo	string	Título original de la película
tituloEspañol	string	Título traducido al español de la película
direccionStreaming	string	Ruta en el servidor de streaming del vídeo
sinopsis	string	Sinopsis de la película
anhoProduccion	int	Año de producción de la película
duracion	int	Duración en minutos de la película
imagen	string	URL de la imagen de la película
actores	string	Actores participan en la película
directores	string	Directores de la película
generoPrincipal	string	Género (cinematográfico) principal de la película
generosSecundarios	string[]	Géneros (cinematográficos) secundarios de la película
paisPrincipal	string	País principal de producción de la película
paisesSecundarios	string[]	Países secundarios de producción de la película
calidadVisual	int	Calidad visual del vídeo (1-5)
anchoVideo	int	Ancho del vídeo
altoVideo	int	Alto del vídeo

Clase: SeccionPrograbados Módulo: grabados

Descripción: Clase encargada de representar información de una sección de programas grabados

Atributo	Tipo	Descripción
idSeccionPrograbados	int	Identificador de la sección de programas grabados
nombreSeccionPograbados	string	Nombre de la sección de programas grabados
imagen	string	URL de la imagen de la sección de programas grabados
descripcion	string	Descripción de la sección de programas grabados

Clase: Prograbado Módulo: grabados

Descripción: Clase encargada de representar información de un programa grabado

Atributo	Tipo	Descripción
idPrograbado	int	Identificador del programa grabado
nombrePrograbado	string	Nombre del programa grabado
cantEmisiones	int	Cantidad de emisiones a visualizar del programa
esNuevo	bool	Indica si es nuevo el programa
descripcion	string	Descripción del programa grabado

Clase: Emision Módulo: grabados

Descripción: Clase encargada de representar información de una emisión de programa grabado

Atributo	Tipo	Descripción
idEmision	int	Identificador de la emisión del programa grabado
direccionStreamming	string	Ruta en el servidor de streaming del vídeo
fechaEmisionTelevision	string	Fecha de emisión en la televisión (formato año-mes-día )
esNueva	bool	Indica si la emisión es nueva
descripcion	string	Descripción de la emisión del programa grabado
calidadVisual	string	Calidad visual del vídeo
anchoVideo	int	Ancho del vídeo
altoVideo	int	Alto del vídeo

Clase: EncuestaMes		Módulo: programados
Descripción: Clase encargada de representar información de una encuesta		
Atributo	Tipo	Descripción
idEncuesta	int	Identificador de la encuesta
coleccionProgramas	ProgramaEncuesta[]	Colección de programas de la encuesta
fechaInicio	string	Fecha de inicio de la encuesta (formato año-mes-día)
FechaFin	string	Fecha de fin de la encuesta (formato año-mes-día)

Clase: ProgramaEncuesta		Módulo: programados
Descripción: Clase encargada de representar información de programa incluido en encuestas		
Atributo	Tipo	Descripción
idPrograma	int	Identificador del programa
NombrePrograma	string	Nombre del programa

Clase: ResultadoEncuestaPorPrograma		Módulo: programados
Descripción: Clase encargada de representar la información de las votaciones por programas		
Atributo	Tipo	Descripción
nombrePrograma	string	Nombre del programa
cantVotos	int	Cantidad de votos realizados al programa

Clase: SeccionEstaciones		Módulo: radioTV
Descripción: Clase encargada de representar la información de las secciones de estaciones de radio o televisión		
Atributo	Tipo	Descripción
idSeccionEstaciones	int	Identificador de las secciones de estaciones
nombreSeccionEstaciones	string	Nombre de la sección de estaciones
imagen	string	URL de la imagen de la sección de estaciones
descripcion	string	Descripción de la sección de estaciones

Clase: Estacion		Módulo: radioTV
Descripción: Clase encargada de representar la información de la estación de radio o televisión		

Atributo	Tipo	Descripción
idEstacion	int	Identificador de la estación
direccionStreaming	string	Ruta del vídeo en el servidor de streaming
imagen	string	URL de la imagen de la estación
descripcion	string	Descripción de la estación

Clase: SeccionSeries		Módulo: series
Descripción: Clase encargada de representar la información de las secciones series		
Atributo	Tipo	Descripción
idSeccionSerie	int	Identificador de las secciones de series
tituloSeccionSeries	string	Título de la sección de series

Clase: Serie		Módulo: series
Descripción: Clase encargada de representar la información de una serie		
Atributo	Tipo	Descripción
idSerie	int	Identificador de la serie
nombreSerie	string	Nombre de la serie
sinopsis	string	Sinopsis de la serie

Clase: Temporada		Módulo: series
Descripción: Clase encargada de representar la información de una temporada de serie		
Atributo	Tipo	Descripción
idTemporada	int	Identificador de la temporada
titulo	string	Título de la temporada
numero	int	Número de la temporada
foto	string	URL de la foto de la temporada
esNueva	bool	Indica si la temporada es nueva

Clase: Capitulo		Módulo: series
Descripción: Clase encargada de representar información de un capítulo de series		
Atributo	Tipo	Descripción
idCapitulo	int	Identificador del capítulo de serie

titulo	string	Título del capítulo
numero	int	Número del capítulo
direccionStreaming	string	Ruta en el servidor de streaming del vídeo
esNuevo	bool	Indica si el capítulo es nuevo
calidadVisual	string	Calidad visual del vídeo
anchoVideo	int	Ancho del vídeo
altoVideo	int	Alto del vídeo

Clase: PeriodoDocente		Módulo: teleclases
Descripción: Clase encargada de representar la información de un período docente (Ejemplo: Primer Semestre, Segundo Semestre, Cursos de Postgrado)		
Atributo	Tipo	Descripción
idPeriodoDocente	int	Identificador del período docente
nombrePeriodoDocente	string	Nombre del período docente

Clase: AnhoDocente		Módulo: teleclases
Descripción: Clase encargada de representar información de un año docente		
Atributo	Tipo	Descripción
idAnho	int	Identificador del año docente
nombreAnhoDocente	string	Nombre del año docente

Clase: Asignatura		Módulo: teleclases
Descripción: Clase encargada de representar información de una asignatura		
Atributo	Tipo	Descripción
idAsignatura	int	Identificador de la asignatura
nombreAsignatura	string	Nombre de la asignatura

Clase: MaterialDocente		Módulo: teleclases
Descripción: Clase encargada de representar información de material docente		
Atributo	Tipo	Descripción
idMaterialDocente	int	Identificador del material docente
tituloMaterial	string	Título del material docente

direccionStreaming	string	Ruta del vídeo en el servidor streaming
sumario	string	Sumario del contenido del material docente
nombreProfesor	string	Nombre del profesor
tipoMaterial	string	Tipo de material (Ejemplo: Conferencia, Clase práctica)
calidadVisual	int	Calidad visual del material (1-5)
anchoVideo	int	Ancho del vídeo
altoVideo	int	Alto del vídeo

### 3.5 Estándar de codificación utilizado

Una de las buenas prácticas propuesta por las metodologías ágiles es la adopción de un estándar de codificación por parte del equipo de desarrollo; asegurando que el código exprese claramente el propósito del mismo y agilice el proceso de refactorización. No se trata de establecer un estándar y que los miembros del equipo lo vean como un régimen, sino que lo apliquen y sobre todo que se sientan cómodos.

Para darle cumplimiento a la solución propuesta se escoge el estándar de código propuesto para el desarrollo de Symfony y se adicionan algunas reglas que hacen más legible el código escrito y con las que el equipo de desarrollo está familiarizado (**Ver Anexo I**).

### 3.6 Conclusiones parciales

Hasta aquí se han descrito los elementos esenciales de la arquitectura del sistema propuesto, los componentes desarrollados y su integración. Se concibe un sistema que cumple con las funcionalidades requeridas por el cliente, flexible ante cambios, seguro y robusto; basado en los patrones de diseño y el patrón arquitectónico que implementa el framework Symfony, unido a los mecanismos de seguridad que provee. Además se describen los servicios web que expone la solución propuesta como interfaces de comunicación; convirtiéndola en una aplicación interoperable, capaz de compartir funcionalidades y datos con otras aplicaciones, independientemente del lenguaje de programación utilizado o plataforma sobre la cual estén desplegadas.

## **Capítulo 4 Validación de la solución propuesta**

Las pruebas que se le realizan a todo producto software son cruciales para ganar en la seguridad y la calidad del mismo, de modo que una vez sea entregado al cliente incurra en el menor número de inconvenientes posibles. Es recomendado realizar pruebas al software desde el mismo momento en que se implementa una nueva funcionalidad, logrando así un entorno de desarrollo satisfactorio. Existen varios tipos de pruebas que pueden ser realizadas a un sistema y que conducen a la obtención de un producto funcional y fiable. Sin embargo, si se trata de desarrollo ágil con pruebas constantes, es interesante destacar la utilización de herramientas que automatizan esta actividad.

### **4.1 Pruebas funcionales**

Las pruebas funcionales son esenciales para determinar si el producto software desarrollado cumple con las funcionalidades requeridas por el cliente. Las mismas validan que la aplicación funcione como un todo; en un ambiente web verifican que todos los enlaces funcionen correctamente, es decir, al hacer clic sobre vínculos y botones, se muestren las páginas y mensajes esperados. Como este proceso se hace manualmente, se torna aburrido y es muy propenso a cometer errores. Es por ello que Symfony incluye su propio framework de pruebas (Lime) además de una clase que representa un navegador web, capaz de automatizar estas tareas (subframework de pruebas funcionales).

Estas pruebas simulan el funcionamiento del navegador, enviando peticiones y recibiendo las respuestas del servidor, tal y como lo haría un usuario del sistema, asegurando que cada funcionalidad haga lo que se supone debe hacer. Es así que los desarrolladores y probadores, a través de la automatización de pruebas funcionales, aseguran la entrega de un producto fiable y que responda de manera esperada ante los requisitos establecidos por el cliente.

#### **4.1.1 Pruebas funcionales realizadas al sistema**

A continuación se describen un conjunto de pruebas diseñadas y ejecutadas para validar la solución propuesta. Fueron realizadas las pruebas a las Historias de Usuario pertenecientes a los módulos cartelera, película, programas grabados y teleclases; que constituyen funcionalidades primordiales para el cliente.

### Caso de Prueba de Aceptación

**Código Caso de Prueba:** I-01-1

**Nombre Historia de Usuario:** Gestionar Cartelera

**Nombre de la persona que realiza la prueba:** Francisco Izquierdo García

**Descripción de la Prueba:** El usuario pulsa sobre el vínculo referente a mostrar asignaturas para un año docente y período docente determinado.

**Condiciones de Ejecución:** Deben existir años docentes y períodos docentes en la base de datos.

**Entrada / Pasos de ejecución:** Implementación y ejecución del caso de prueba.

**Resultado Esperado:** Muestra listado de asignaturas del año docente y período docente especificado.

**Evaluación de la Prueba:** Satisfactoria

```
frizquierdo@Internos:/var/www/francisco/test$ php symfony test:functional internos carteleraActions
> Chequeando lista de asignaturas por año y período docente
# get /teleclase/abrirAnho?id_periodo=1&id_anho=1
ok 1 - request parameter module is teleclase
ok 2 - request parameter action is abrirAnho
ok 3 - request parameter id_periodo is 1
ok 4 - request parameter id_anho is 1
ok 5 - status code is 200
1..5
Looks like everything went fine.
```

### Caso de Prueba de Aceptación

**Código Caso de Prueba:** I-01-2

**Nombre Historia de Usuario:** Gestionar Cartelera

**Nombre de la persona que realiza la prueba:** Francisco Izquierdo García

**Descripción de la Prueba:** El usuario trata de eliminar un canal que no existe.

**Condiciones de Ejecución:** No existe un canal con el identificador especificado en la base de datos

**Entrada / Pasos de ejecución:** Implementación y ejecución del caso de prueba.

**Resultado Esperado:** El sistema muestra una página de error 404

**Evaluación de la Prueba:** Satisfactoria

```
frizquierdo@Internos:/var/www/francisco/test$ php symfony test:functional internos carteleraActions
> Chequeando eliminar canal dado id de canal no existente
# get /cartelera/eliminarCanal?id_canal=1
ok 1 - request parameter module is cartelera
ok 2 - request parameter action is eliminarCanal
ok 3 - request parameter id_canal is 1
ok 4 - status code is 404
1..4
Looks like everything went fine.
```

## Caso de Prueba de Aceptación

**Código Caso de Prueba:** I-02-1

**Nombre Historia de Usuario:** Gestionar sección de películas

**Nombre de la persona que realiza la prueba:** Francisco Izquierdo García

**Descripción de la Prueba:** El usuario trata de insertar una película en una sección que no existe.

**Condiciones de Ejecución:** No existen secciones de películas o la indicada no existe en la base de datos.

**Entrada / Pasos de ejecución:** Implementación y ejecución del caso de prueba.

**Resultado Esperado:** El sistema muestra un mensaje de error indicando no existe la sección de película.

**Evaluación de la Prueba:** Satisfactoria

```
frizquierdo@Internos:/var/www/francisco/test$ php symfony test:functional internos peliculaActions
> Insertar Pelicula en una sección que no existe
# post /pelicula/crearPelicula?tpelicula[id_seccion_pelicula]=20
ok 1 - request method is POST
ok 2 - request parameter module is pelicula
ok 3 - request parameter action is crearPelicula
ok 4 - request parameter tpelicula[id_seccion_pelicula] is 20
ok 5 - status code is 406
1..5
Looks like everything went fine.
```

### Caso de Prueba de Aceptación

**Código Caso de Prueba:** I-06-1

**Nombre Historia de Usuario:** Gestionar Programas Grabados

**Nombre de la persona que realiza la prueba:** Francisco Izquierdo García

**Descripción de la Prueba:** El usuario pulsa sobre el enlace referentes a mostrar los programas de una sección de programas grabados.

**Condiciones de Ejecución:** Debe de existir la sección de programas grabados en la base de datos y además existen programas para esa sección.

**Entrada / Pasos de ejecución:** Implementación y ejecución del caso de prueba.

**Resultado Esperado:** El sistema muestra un listado de programas pertenecientes a la sección.

**Evaluación de la Prueba:** Satisfactoria

```
frizquierdo@Internos:/var/www/francisco/test$ php symfony test-functional internos seccionPrograbadosActions
> Chequeando lista de programas grabados dado id de sección de programas
# get /seccionPrograbados/mostrarProgramas?id_seccion_prograbados=1
ok 1 - request parameter module is seccionPrograbados
ok 2 - request parameter action is mostrarProgramas
ok 3 - request parameter id_seccion_prograbados is 1
ok 4 - status code is 200
ok 5 - response selector tr.tr_impar matches 2 times
1..5
Looks like everything went fine.
```

### Caso de Prueba de Aceptación

**Código Caso de Prueba:** I-06-2

**Nombre Historia de Usuario:** Gestionar Programas Grabados

**Nombre de la persona que realiza la prueba:** Francisco Izquierdo García

**Descripción de la Prueba:** El usuario pulsa sobre el enlace referente a eliminar un programa grabado.

**Condiciones de Ejecución:** Debe de existir el programa grabado en la base de datos

**Entrada / Pasos de ejecución:** Implementación y ejecución del caso de prueba.

**Resultado Esperado:** El sistema muestra la lista de programas grabados actualizada.

**Evaluación de la Prueba:** Satisfactoria

```
frizquierdo@Internos:/var/www/francisco/test$ php symfony test:functional internos seccionPrograbadosActions
> Eliminar programa grabado dado id de programa
# get /prograbados/eliminarPrograma?id_prograbado=2
ok 1 - request parameter module is prograbados
ok 2 - request parameter action is eliminarPrograma
ok 3 - request parameter id_prograbado is 2
ok 4 - page redirected to http://localhost/index.php/prograbados
ok 5 - status code is 302
1..5
Looks like everything went fine.
```

### Caso de Prueba de Aceptación

**Código Caso de Prueba:** I-06-3

**Nombre Historia de Usuario:** Gestionar Programas Grabados

**Nombre de la persona que realiza la prueba:** Francisco Izquierdo García

**Descripción de la Prueba:** El usuario pulsa sobre el enlace referente a mostrar las emisiones de un programa grabado

**Condiciones de Ejecución:** Debe de existir el programa grabado en la base de datos

**Entrada / Pasos de ejecución:** Implementación y ejecución del caso de prueba.

**Resultado Esperado:** El sistema muestra la lista de emisiones referentes al programa grabado.

**Evaluación de la Prueba:** Satisfactoria

```
frizquierdo@Internos:/var/www/francisco/test$ php symfony test-functional internos seccionPrograbadosActions
> Chequeando lista de emisiones dado id de programa al que pertenecen
# get /prograbados/mostrarEmisiones?id_prograbado=3
ok 1 - request parameter module is prograbados
ok 2 - request parameter action is mostrarEmisiones
ok 3 - request parameter id_prograbado is 3
ok 4 - status code is 200
ok 5 - response selector tr.tr_impar matches 1 times
1..5
Looks like everything went fine.
```

### Caso de Prueba de Aceptación

**Código Caso de Prueba:** I-08-1

**Nombre Historia de Usuario:** Gestionar Teleclases.

**Nombre de la persona que realiza la prueba:** Francisco Izquierdo García

**Descripción de la Prueba:** El usuario intenta insertar un período docente con datos incorrectos en el formulario.

**Condiciones de Ejecución:** Formulario con datos incorrectos.

**Entrada / Pasos de ejecución:** Implementación y ejecución del caso de prueba.

**Resultado Esperado:** El sistema muestra mensaje de error en el formulario.

**Evaluación de la Prueba:** Satisfactoria

```
frizquierdo@Internos:/var/www/francisco/test$ php symfony test:functional internos teleclaseActions
> Adicionar Período Docente con error en los datos del formulario
# post /teleclase/crearPeriodo?tperiodo_docente[nombre_periodo_docente]=
ok 1 - request method is POST
ok 2 - request parameter module is teleclase
ok 3 - request parameter action is crearPeriodo
ok 4 - status code is 406
1..4
Looks like everything went fine.
```

## Caso de Prueba de Aceptación

**Código Caso de Prueba:** I-08-2

**Nombre Historia de Usuario:** Gestionar Teleclases.

**Nombre de la persona que realiza la prueba:** Francisco Izquierdo García

**Descripción de la Prueba:** El usuario pulsa sobre el enlace referente a mostrar los años docentes que tienen asignatura dentro de un período determinado.

**Condiciones de Ejecución:** Debe de existir el período y año docente, y además tener registrada al menos una asignatura.

**Entrada / Pasos de ejecución:** Implementación y ejecución del caso de prueba.

**Resultado Esperado:** El sistema muestra listado de años docentes que tienen asignaturas en el período docente dado.

**Evaluación de la Prueba:** Satisfactoria

```
frizquierdo@Internos:/var/www/francisco/test$ php symfony test-functional internos teleclaseActions
> Chequeando lista de años docentes con asignaturas para un período dado
# get /teleclase/abrirPeriodo?id_periodo=1
ok 1 - request parameter module is teleclase
ok 2 - request parameter action is abrirPeriodo
ok 3 - request parameter id_periodo is 1
ok 4 - status code is 200
ok 5 - response selector tr.tr_impar matches 1 times
1..5
Looks like everything went fine.
```

## 4.2 Impacto y aporte de la solución propuesta

Toda investigación o producto que tenga como objetivo resolver determinado problema generalmente debe tener un impacto, ya sea para la sociedad o en el aspecto económico. El impacto que produce y el aporte que haga al desarrollo de la humanidad demuestra su importancia. La presente investigación al término de su finalización obtuvo una aplicación web libre que automatiza las tareas de publicación de los materiales de los diferentes espacios del portal Inter-Nos, flexible y adaptable ante cambios y lo suficientemente escalable. El sistema permite la gestión de series, películas, programas grabados, carteleras de programas en televisión, teleclases y estaciones de radio y televisión. Esta gestión incluye tareas de publicación de los diferentes materiales audiovisuales a través de calendarios automatizados; además posibilita la gestión de los protocolos de streaming, entre otras funcionalidades. Con el uso de la aplicación obtenida se humaniza el trabajo de los administradores del portal, ya que son innecesarios conocimientos de programación web o administración de bases de datos para la publicación de los materiales.

Además de las tareas de publicación, la aplicación brinda un conjunto de servicios web a través de los cuales el sistema es capaz de compartir datos y funcionalidades con otras aplicaciones,

independientemente de su lenguaje de implementación o sistema operativo en el que están desplegadas. Estos servicios web pueden ser consumidos por cualquier aplicación desarrollada dentro de la comunidad universitaria y el portal Inter-Nos, característica que está a tono con la política de la Dirección de Informatización en cuanto al desarrollo de aplicaciones en la Universidad de las Ciencias Informáticas hacia una Arquitectura Empresarial basada en servicios. De esta manera la solución alcanzada más allá de ser una aplicación monolítica y cerrada, se convierte en un sistema interoperable y horizontal, al interactuar con otras aplicaciones de la intranet universitaria. En el aspecto económico el desarrollo de esta solución contribuye al ahorro de un total de €15,000 (equivalente a \$22,000 USD) por año en la adquisición del software **Magnolia Enterprise Edition Pro**, un producto que como fue planteado no cumplía con los requerimientos del cliente; y además ahorra un total de €8,000 (equivalente a \$12,000 USD) por año en la compra de la versión empresarial no profesional (versión estándar) del mismo producto.

### **4.3 Conclusiones parciales**

Mediante la automatización de las pruebas funcionales que fueron realizadas al sistema se agiliza el proceso de desarrollo, influyendo en la rápida corrección de errores en las funcionalidades implementadas, y que fuesen validados todos los caminos lógicos del código escrito con un mínimo margen de error en las pruebas realizadas. Se garantiza entonces la calidad de la aplicación propuesta que hace lo que se supone debe hacer.

## Conclusiones

El avance acelerado de la ciencia y las tecnologías, y su aplicación en todos los ámbitos del desarrollo de la humanidad hacen posible el uso de sistemas informáticos como el que se logra al finalizar la presente investigación. Durante el desarrollo de la misma se han ejecutado una serie de pasos lógicos que tributan a la obtención del resultado esperado.

Fueron analizadas algunas soluciones de gestión de contenidos audiovisuales para identificar y documentar sus características esenciales, estudio que permitió determinar la necesidad de desarrollar la solución propuesta capaz de automatizar los procesos de gestión de los materiales audiovisuales del portal Inter-Nos, y fuese lo suficientemente escalable y adaptable ante cambios. Se analizaron los diferentes aspectos relacionados con el diseño e implementación de servicios web, cuestiones que posibilitaron apreciar el papel importante que juegan en la comunicación entre sistemas.

Además el análisis efectuado sobre las tecnologías utilizadas en la creación de servicios web, permitió la implementación de un sistema capaz de compartir datos y funcionalidades con otras aplicaciones. Con el uso de las herramientas, tecnologías y lenguajes de programación libres que soportaron el proceso de desarrollo, el cual estuvo guiado por la metodología SXP, se obtiene un producto completamente libre, a través de un ciclo de desarrollo ágil, permitiendo enfrentarse a cambios en los requisitos. El diseño y ejecución de un conjunto de pruebas funcionales realizadas al culminar cada nueva funcionalidad aseguraron la calidad del producto final.

Fueron documentadas las funcionalidades que cumple el sistema para su posterior implementación; proceso del que se obtiene un producto software que permite la gestión eficiente de los materiales publicados en los diferentes espacios del portal Inter-Nos, a través de la automatización de las tareas de publicación de los materiales audiovisuales. Esta información gestionada desde el sistema es expuesta mediante servicios web disponibles que pueden ser consumidos por el portal Inter-Nos y la comunidad universitaria.

## Recomendaciones

Al concluir la presente investigación, se recomienda:

- Agregar nuevos módulos que gestionen los espacios Eureka, Panorama Político y la portada del portal Inter-Nos.
- Que incorpore nuevos procesos de gestión de medias, por ejemplo, la búsqueda de materiales audiovisuales.
- Incorporación de nuevos servicios web que contribuyan con la búsqueda de materiales desde el portal Inter-Nos u otra aplicación que consuma los servicios.
- Incorpore a los procesos de catalogación de los archivos de vídeo, la gestión de la información técnica de los materiales a través de metadatos.

## Referencias Bibliográficas

- [1] D.S.R. López, "INTERNET: NUEVO MEDIO DE DIFUSIÓN PARA LOS CONTENIDOS TELEVISIVOS," vol. V1, Dec. 2006, pp. 55-64.
- [2] A.V. Gómez, "Servicio de Media Streaming para la Web. Portal Inter-nos. Módulos: Teleclases y TV," Universidad de las Ciencias Informáticas, 2006.
- [3] Rolando Alfredo Hernández León and Zaida Coello González, El Paradigma Cuantitativo de la Investigación Científica, Universidad de las Ciencias Informáticas: Editorial Universitaria (EDUNIV), 2002.
- [4] "ci glossary."
- [5] Departamento de Técnicas de Programación, "Introducción a la Programación cliente-servidor. Aplicaciones Web. Funcionamiento de un servidor Web," 2009.
- [6] M.A. Álvarez, "Qué es un CMS," Nov. 2008.
- [7] D.P. Roldán, "CMS y Foros. Curso de Herramientas Colaborativas," 2007.
- [8] ISID, "ISID Gestión Inteligente de Activos Multimedia," 2009.
- [9] "Magnolia On Air - Products - Magnolia CMS – Simple Open Source Content Management," 2008.
- [10] B. Hüseman, "OntoMedia - semantic multimedia management for private users," May. 2005.
- [11] Distributed Multimedia Information System Laboratory, "FIU - The Distributed Multimedia Information System Laboratory."
- [12] J. Eguiluz, "ApplicationsDevelopedWithSymfony - symfony - Trac," 2010.
- [13] GET Movie, "GET Movie," 2008.
- [14] Elmulticine.com, "Cine películas estrenos cartelera. 2009," 2009.
- [15] Sensacine, "SensaCine.com: Cine, Cartelera, Estrenos de Cine, Series TV, Películas, Tráilers. 2009," 2009.
- [16] Departamento de Ingeniería de Software, UCI, "Introducción a la Ingeniería de Software."
- [17] K. Beck and M. Fowler, "Principles behind the Agile Manifesto."
- [18] A.M. Abad and Gladys Marsi Peñalver Romero, "SXP, metodología ágil para proyectos de software libre," Universidad de las Ciencias Informáticas, 2009.

- [19] <http://www.docs.php.net>, "PHP: Historia de PHP - Manual," Sep. 2009.
- [20] O. Ben-Kiki and Clark Evans, "The Official YAML Web Site."
- [21] Doctrine Project, "Doctrine ORM for PHP."
- [22] W3C, "Cascading Style Sheets."
- [23] J.J. Garrett, "adaptive path » ajax: a new approach to web applications," Feb. 2005.
- [24] F. Potencier and F. Zaninotto, "Capítulo #1. Introducción a Symfony," p. 7.
- [25] Learning Ex JS Building dynamic, desktop-style user interfaces for your data-driven web applications, Packt Publishing Ltd., 2008.
- [26] Ellislab Inc, "Welcome to CodeIgniter : CodeIgniter User Guide," 2009.
- [27] A. Gutiérrez and D. Teejda, "Kumbia PHP Framework. Por qué programar debería ser más fácil."
- [28] Inc. Cake Software Foudnation, "El manual :: 1.2 Colección :: El Cookbook," 2009.
- [29] Ltd. Zend Technologies, "Zend Framework," 2009.
- [30] F. Potencier and F. Zaninotto, "Capítulo #1. Introducción a Symfony," Symfony 1.2, la guía definitiva, pp. 7-8.
- [31] PostgreSQL Global Development Group, "PostgreSQL: History," 2009.
- [32] C. Mateu, Desarrollo de aplicaciones web, Eureka Media., 2004.
- [33] Mohammed J. Kabir, La Bilbia del Servidor Apache 2, Anaya Multimedia, 2003.
- [34] Compañía Ibermática, "SOA - Web Services — ibermatica.com."
- [35] R. Richards, "Introducing the Web Services Description Language (WSDL)," Pro PHP XML and Web Services, Apress, 2006, p. 673.
- [36] R. Richards, "Universal Description, Discovery and Integration (UDDI)," Pro PHP XML and Web Services, Apress, 2006, p. 752.
- [37] R. Richards, "Web Distributed Data Exchange, XML-RPC, Representational State Transfer (REST), SOAP," Pro PHP XML and Web Services, Apress, 2006, pp. 567-750.
- [38] Dirección de Informatización, Universidad de las Ciencias Informáticas, "Arquitectura para los Sistemas que Conforman la Intranet Universitaria," May. 2007.
- [39] C. Alexander, S. Ishikawa, and M. Silverstein, A Pattern Language, Oxford University Press: 1977.
- [40] E. Gamma, R. Helm, R. Johnson, and J. Vlissides, "Organizing the Catalog," Design Patterns: Elements of Reusable Objects-Oriented Software, Addison Wesley, 1995, pp. 21-22.
- [41] F. Potencier and F. Zaninotto, "Capítulo 2. Explorando el interior de Symfony," Symfony 1.2, la

guía definitiva, p. 18.

- [42] M. Fowler, "Web Server Patterns. Model View Controller," *Patterns of Enterprise Application Architecture*, Pearson Education, Inc, 2003, pp. 330-331.
- [43] F. Potencier and F. Zaninotto, "Capítulo 2. Explorando el interior de Symfony," *Symfony 1.2, la guía definitiva*, p. 26.
- [44] J. Rumbaugh, I. Jacobson, and G. Booch, "Vistas Físicas," *El Lenguaje Unificado de Modelado. Manual de Referencia*, Addison Wesley, .

## Bibliografía

- [1] J.J. Garrett, "adaptive path » ajax: a new approach to web applications," Feb. 2005.
- [2] C. Alexander, S. Ishikawa, and M. Silverstein, A Pattern Language, Oxford University Press: 1977.
- [3] J. Eguiluz, "ApplicationsDevelopedWithSymfony - symfony - Trac," 2010.
- [4] Dirección de Informatización, Universidad de las Ciencias Informáticas, "Arquitectura para los Sistemas que Conforman la Intranet Universitaria," May. 2007.
- [5] F. Potencier and F. Zaninotto, "Capítulo #1. Introducción a Symfony," p. 7.
- [6] F. Potencier and F. Zaninotto, "Capítulo #1. Introducción a Symfony," Symfony 1.2, la guía definitiva, pp. 7-8.
- [7] F. Potencier and F. Zaninotto, "Capítulo 2. Explorando el interior de Symfony," Symfony 1.2, la guía definitiva, p. 26.
- [8] F. Potencier and F. Zaninotto, "Capítulo 2. Explorando el interior de Symfony," Symfony 1.2, la guía definitiva, p. 18.
- [9] W3C, "Cascading Style Sheets."
- [10] "ci glossary," Jan. 2010.
- [11] Elmulticine.com, "Cine películas estrenos cartelera. 2009," 2009.
- [12] D.P. Roldán, "CMS y Foros. Curso de Herramientas Colaborativas," 2007.
- [13] C. Mateu, Desarrollo de aplicaciones web, Eureka Media., 2004.
- [14] Doctrine Project, "Doctrine ORM for PHP."
- [15] Inc. Cake Software Foudnation, "El manual :: 1.2 Colección :: El Cookbook," 2009.
- [16] Rolando Alfredo Hernández León and Zaida Coello González, El Paradigma Cuantitativo de la Investigación Científica, Universidad de las Ciencias Informáticas: Editorial Universitaria (EDUNIV), 2002.
- [17] Distributed Multimedia Information System Laboratory, "FIU - The Distributed Multimedia Information System Laboratory."
- [18] GET Movie, "GET Movie," 2008.
- [19] "GForge-Facultad10: SXP: Información del Proyecto."
- [20] D.S.R. López, "INTERNET: NUEVO MEDIO DE DIFUSIÓN PARA LOS CONTENIDOS TELEVISIVOS," vol. V1, Dec. 2006, pp. 55-64.
- [21] Departamento de Ingeniería de Software, UCI, "Introducción a la Ingeniería de Software."
- [22] Departamento de Técnicas de Programación, "Introducción a la Programación cliente-servidor.

- [23] R. Richards, “Introducing the Web Services Description Language (WSDL),” Pro PHP XML and Web Services, Apress, 2006, p. 673.
- [24] ISID, “ISID Gestión Inteligente de Activos Multimedia,” 2009.
- [25] A. Gutiérrez and D. Teejda, “Kumbia PHP Framework. Por qué programar debería ser más fácil..”
- [26] Mohammed J. Kabir, La Biblia del Servidor Apache 2, Anaya Multimedia, 2003.
- [27] Learning Ex JS Building dynamic, desktop-style user interfaces for your data-driven web applications, Packt Publishing Ltd., 2008.
- [28] “Magnolia On Air - Products - Magnolia CMS – Simple Open Source Content Management,” 2008.
- [29] B. Hüseman, “OntoMedia - semantic multimedia management for private users,” May. 2005.
- [30] E. Gamma, R. Helm, R. Johnson, and J. Vlissides, “Organizing the Catalog,” Design Patterns: Elements of Reusable Objects-Oriented Software, Addison Wesley, 1995, pp. 21-22.
- [31] <http://www.docs.php.net>, “PHP: Historia de PHP - Manual,” Sep. 2009.
- [32] PostgreSQL Global Development Group, “PostgreSQL: History,” 2009.
- [33] K. Beck and M. Fowler, “Principles behind the Agile Manifesto.”
- [34] M.A. Álvarez, “Qué es un CMS,” Nov. 2008.
- [35] Sensacine, “SensaCine.com: Cine, Cartelera, Estrenos de Cine, Series TV, Películas, Tráilers. 2009,” 2009.
- [36] A.V. Gómez, “Servicio de Media Streaming para la Web. Portal Inter-nos. Módulos: Teleclases y TV,” Universidad de las Ciencias Informáticas, 2006.
- [37] Compañía Ibermática, “SOA - Web Services — ibermatica.com.”
- [38] A.M. Abad and Gladys Marsi Peñalver Romero, “SXP, metodología ágil para proyectos de software libre,” Universidad de las Ciencias Informáticas, 2009.
- [39] O. Ben-Kiki and Clark Evans, “The Official YAML Web Site.”
- [40] R. Richards, “Universal Description, Discovery and Integration (UDDI),” Pro PHP XML and Web Services, Apress, 2006, p. 752.
- [41] J. Rumbaugh, I. Jacobson, and G. Booch, “Vistas Físicas,” El Lenguaje Unificado de Modelado. Manual de Referencia, Addison Wesley, .
- [42] R. Richards, “Web Distributed Data Exchange, XML-RPC, Representational State Transfer (REST), SOAP,” Pro PHP XML and Web Services, Apress, 2006, pp. 567-750.
- [43] M. Fowler, “Web Server Patterns. Model View Controller,” Patterns of Enterprise Application Architecture, Pearson Education, Inc, 2003, pp. 330-331.
- [44] Ellislab Inc, “Welcome to CodeIgniter : CodeIgniter User Guide,” 2009.

[45] F. Potencier and F. Zaninotto, "YAML," *Symfony 1.2, la guía definitiva*, 2008, p. 16.

[46] Ltd. Zend Technologies, "Zend Framework," 2009.

### Anexo I Estándar de codificación utilizado

1. Nunca use tabulación en el código. La indentación está hecha por pasos de 2 espacios:

```
<?php
    class sfFoo
    {
        public function bar()
        {
            sfCafe::hacer();
        }
    }
}
```

2. No ponga espacios después de abrir un paréntesis ni antes de cerrarlo:

```
<?php
    if ($miVariable == getRequestValue($name)) // correcto
    if ( $miVariable == getRequestValue($name) ) // incorrecto
```

3. Use la notación camelCase, no guión bajo, para variables, funciones y nombre de métodos:

```
function hacerCafe() // correcto
function HacerCafe() // incorrecto
function hacer_cafe() // incorrecto
```

4. Los miembros públicos de las clases actions que tengan una vista asociada se les agregará el prefijo execute:

```
public function executeInicio()
public function executeCrear()
public function executeNuevo()
public function executeEditar()
public function executeActualizar()
public function executeEliminar()
```

5. Use guión bajo para opciones/argumentos/nombre de parámetros.

6. Las llaves siempre irán en su propia línea.

7. Use llaves para indicar el cuerpo de las estructuras de control, sin tener en cuenta el número de sentencias que éstas contengan.

8. Symfony está escrito en PHP5, por lo que cada método de una clase o definición de miembro

puede declarar su visibilidad explícitamente usando las palabras reservadas `private`, `protected` ó `public`.

9. No termine el código de los archivos de librerías con la etiqueta de cierre `?>`. Esto es porque realmente no es necesario y puede traer problemas en las salidas si en algún momento tiene un espacio en blanco al final de dicha etiqueta.

10. En el cuerpo de las funciones, la sentencia `return` debe tener una línea en blanco a priori para incrementar la legibilidad:

```
<?php
function hacerCafe()
{
    if (false !== estaDurmiendo() && false !== tieneSuficienteCafeinaPorHoy())
    {
        puedeHacerCafe();

        return 1;
    }
    else
    {
        noPuedeHacerCafe();
    }

    return null;
}
```

11. Todo comentario debe estar en sus propias líneas y con el siguiente formato:

```
<?php
    // primero un espacio, no necesita etiqueta de cierre.
```

12. Evite la evaluación de variables dentro de una cadena, en cambio use la concatenación:

```
<?php
$string = 'algo';
$nuevoString = "$string es imponente!"; // mal
$nuevoString = $string.' es imponente!'; // bien
$nuevoString = sprintf('%s es imponente', $string);
// para mensajes de excepción y cadenas con muchas sustituciones.
```

13. Use minúsculas para constantes tipadas nativas de PHP: `false`, `true` y `null`. Lo mismo para `array()`. Todo lo contrario, siempre use cadenas en mayúsculas para la definición de sus propias constantes, como `define('MI_CONSTANTE', 'foo/bar')`. Mejor, trate de usar siempre las constantes de clases:

```
<?php
class sfCafe
{
    const TIENE_AZUCAR = true;
}
var_dump(sfCafe::TIENE_AZUCAR);
```

14. Cuando compare una variable con un string, ponga primero el string y luego use un tipo de prueba siempre y cuando sea aplicable:

```
<?php
if ("cadena" === $variable)
```

15. Use tipos de PHP recalando en las firmas de los métodos y funciones:

```
<?php
public function notificar(sfEvent $event)
{
    // ...
}
```

16. Todas las funciones y métodos de clases deben tener su propio bloque de documentación (phpdoc). Las sentencias @ no terminan en punto final:

@param indica el tipo y nombre de la variable. Si la variable puede tener múltiples tipos, entonces un tipo mixto debe ser usado.

Lo ideal es que los bloques @... estén alineados verticalmente (con espacios).

```
<?php
/**
 * Notifica a todos los oyentes de un evento determinado.
 *
 * @param sfEvent $evento sfEvent instancia
 *
 * @return sfEvent      sfEvent instancia
 */
public function notificar(sfEvent $event)
```

17. Los nombres de las tablas de la base de datos serán lo más descriptivo posible, iniciarán con el prefijo t, y para el caso de nombres compuestos se usará guión bajo:

- tseccion\_series
- tpeliculas
- tcapitulos

18. Los nombres de los atributos de las tablas de la base de datos serán lo más descriptivos posibles, usando guión bajo para nombres compuestos por más de una palabra y siempre en minúscula:

- id\_seccion\_serie
- titulo\_pelicula



## Glosario de Términos

**Accesibilidad:** de fácil disponibilidad y acceso. Es el grado con que cualquier persona puede acceder a determinado servicio u objeto, independientemente de las capacidades físicas, técnicas o cognitivas que posea.

**Ancho de Banda:** en el campo de la Computación de Redes y Ciencias de la Computación, también conocido como capacidad de ancho de banda o ancho de banda disponible en bit/s, es el rango máximo de transferencia de datos en un enlace de comunicación digital.

**ASP (Active Server Page):** lenguaje de programación web desarrollado por la compañía Microsoft, que permite la creación de páginas web dinámicas a través de un conjunto de scripts interpretados del lado del servidor.

**Cookies:** son ficheros que se almacenan en las máquinas clientes conteniendo información pasada desde los sitios web y que es usada para futuras vistas al mismo sitio.

**Escalabilidad:** en Ingeniería Informática y redes se refiere a la cualidad que posee un sistema informático de crecer en servicios ofrecidos sin perder calidad o responder a nuevos flujos de trabajo o circunstancias cambiantes de manera eficiente.

**Helpers:** funciones simples que encapsulan grandes bloques de código con el fin de hacer más legible y sencillo su uso.

**Hipertextos:** conjunto estructurado de textos, gráficos y otros elementos enlazados entre sí.

**Historias de Usuarios:** técnica utilizada en la metodología de desarrollo de software XP, para especificar los requisitos del software.

**Ingeniería de Software:** es una tecnología multicapas, en la que se identifican métodos, procesos y herramientas. Es la aplicación de un enfoque cuantificable, disciplinado y sistemático al desarrollo, operación y mantenimiento del software.

**Internet:** conjunto de redes descentralizadas de ordenadores interconectados entre sí, que comparten diversos servicios de comunicación.

**Interoperabilidad:** es la capacidad de los sistemas de intercambiar procesos y datos, comúnmente a

través de servicios.

**Intranet:** una red de ordenadores privados, que proporciona los servicios de red necesarios a nivel de empresa u organización.

**JSON (JavaScript Object Notation):** formato de intercambio de datos ligero. Independiente del lenguaje de programación, representa un subconjunto de la notación de objetos JavaScript.

**Metadato:** datos que describen otros datos.

**MSSQL Server 2000 (Microsoft SQL Server 2000):** sistema gestor de bases de datos relacionales desarrollado por la compañía Microsoft.

**Open Source (código abierto):** un código fuente de un programa informático que permite determinadas libertades sobre el mismo.

**ORM (Object Relational Mapping):** una técnica para representar las tablas de bases de datos relacionales y sus relaciones como clases de un lenguaje orientado a objeto.

**PDF (Portable Document Format):** formato de archivos desarrollado por Adobe Systems.

**PHPUnit:** es un framework para el desarrollo automatizado de pruebas unitarias para el lenguaje PHP.

**Plugin o plug in (complemento):** es una aplicación que se integra como componente de otra para aportarle una nueva funcionalidad.

**Portabilidad:** en términos informáticos se refiere a la independencia que posee el código fuente de un sistema informático del sistema operativo que lo utilice, constituyendo un código reusable en varios sistemas operativos.

**Prototipo de interfaz de usuario no funcional:** representación limitada del diseño de un producto que permite a las partes responsables de su creación experimentar su uso, probarlo en situaciones reales y explorar su uso. Un prototipo puede ser cualquier cosa, desde un trozo de papel con sencillos dibujos a un complejo software.

**Refactorización:** reestructuración y reutilización de un código previamente escrito.

**Release:** se refiere a una versión funcional de un producto software.

**Repositorio de Contenido:** sitio centralizado en el que se gestiona y almacena información digital.

**Scaffolding (andamiaje):** una técnica que permite a un desarrollador definir y crear aplicaciones básicas que pueden crear, retirar, actualizar y borrar objetos.

**SCRUM:** define un marco de trabajo ágil para la gestión de proyectos, basado en iteraciones y entregas incrementales de desarrollo de un producto o servicio. Especialmente indicado para proyectos con requisitos cambiantes.

**Servicio Web:** es un sistema software diseñado para soportar la interacción máquina-máquina a través de una red, utilizando protocolos estándares de Internet y XML como formato de mensajes.

**Servidor Web:** software que suministra páginas Web en respuesta a las peticiones de los navegadores Web.

**SmallTalk:** lenguaje de programación, desarrollado en los años 70, fue uno de los primeros en incluir el término orientación a objetos, aunque no con el mismo significado de los lenguajes actuales como C++, JAVA, C# o PHP.

**Software:** programas del ordenador. Se refiere a instrucciones que se ejecutan por medio de un equipo de cómputo.

**Streaming:** Tecnología de distribución de vídeo bajo demanda o en vivo a través de Internet. Los contenidos distribuidos son leídos, examinados y procesados mientras se van descargando, de forma que es posible reproducir estos contenidos mientras el resto se va descargando.

**Tooltips (descripción emergente):** mensajes emergentes asociados a componentes visuales, con el objetivo de mostrar información referente al componente.

**UML (Unified Modeling Language):** Lenguaje Unificado de Modelado, es un lenguaje de modelado de sistemas de software. Permite la especificación, visualización, construcción y documentación de elementos de la Ingeniería del Software.

**W3C (World Wide Web Consortium):** organismo internacional que tiene por fin establecer normas para el desarrollo y uso de la web.

**Web Semántica:** es la descripción de los datos a través de metadatos (datos que describen y ayudan a ubicar otros datos) llevado al marco de la web. La información queda reflejada a través de otros

datos que cobran sentido y significado, de manera que las máquinas pueden procesarla.

**XP (eXtremme Programing):** es una metodología ágil centrada en potenciar las relaciones interpersonales como clave para el éxito en desarrollo de software, promoviendo el trabajo en equipo, preocupándose por el aprendizaje de los desarrolladores, y propiciando un buen clima de trabajo.