



**Universidad de las Ciencias Informáticas
Facultad 8**

TRABAJO DE DIPLOMA PARA OPTAR POR EL TÍTULO DE INGENIERO EN
CIENCIAS INFORMÁTICAS

**Sistema de Gestión de Información del Polo de
Software Educativo y Multimedia**

Autores:

Ernesto Muñoz Castillo
Yan Monteagudo Jiménez

Tutores:

Ing. Yasirys Terry González
Ing. Osdalme Fuentes Colina

DECLARACIÓN DE AUTORÍA

Declaramos ser autores de la tesis titulada Sistema de Gestión de Información del Polo de Software Educativo y Multimedia 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 _____.

Firma del Autor

Yan Monteagudo Jiménez

Firma del Autor

Ernesto Muñoz Castillo

Firma del Tutor

Ing. Yasirys Terry González

Firma del Cotutor

Ing. Osdalme Fuentes Colina

PENSAMIENTO

Solo los cristales se rompen, los hombres mueren de pie.

A handwritten signature in black ink, appearing to be the name 'J. L.' or similar, written in a cursive style.

DEDICATORIA

...A mi familia por todo el apoyo, dedicación y confianza que siempre han depositado en mí. Sin ustedes no hubiera podido lograr lo que he sido hasta hoy...

Ernesto

...A mis padres, por ser mis amigos, mis guías, por ser el faro que ilumina mi vida.

A mi esposa, por ser el diamante del cofre de mi corazón.

A mi niño que es mi razón de ser.

A mi segunda familia, por estar siempre presente y darme la confianza en mí mismo para seguir adelante.

A mis hermanos, por dejarme ser parte de sus vidas, a pesar de la distancia...

Yan

AGRADECIMIENTOS

...A mis padres, mis abuelos, mis hermanos, primos y tíos.

...A mis amigos.

...A mis tutores por la paciencia y ayuda durante todo el transcurso de la tesis.

...A todas aquellas personas que me ayudaron en el trayecto de mi carrera, aguantaron todas mis malcriadeces y que a pesar de todo siempre confiaron en mí.

Ernesto

...A mis padres Ramón y Lidia por estar siempre en el momento que los necesité, por depositar toda su confianza en mi, por brindarme todo su apoyo en todos estos años de estudio.

...A mi esposa Litzandra por ser el único diamante del cofre de mi corazón, por ser mi vida, mi luz, mi guía y haber compartido un momento tan lindo como el nacimiento de nuestro niño Alejandro.

...A mi segunda familia Pedro y Olga por todo su apoyo y cariño incondicional, por haber sido como unos padres para mí.

...A toda mi familia y muy especial a mis hermanos Miguel Liban, Liuba y Lázaro por llevarme presente y darme seguridad en lo que hacía a pesar de la distancia.

...A mis tutores Yasirys y Osdalme por toda su paciencia y apoyo en estos meses.

...A todas aquellas personas que de una forma u otra se vieron afectadas por mi en algún momento de la vida.

...A todos mis amigos y compañeros de grupo, por estar siempre cuando los necesité.

Yan

RESUMEN

En la Facultad 8, el Polo de Software Educativo y Multimedia requiere un control estricto de su desempeño, pues en el mismo se llevan a cabo diferentes procesos que han progresado paulatinamente. Debido a lo antes mencionado es necesario elaborar un sistema con la finalidad de automatizar el control y gestión de la información generada por estos procesos. El presente trabajo se ha centralizado en el análisis, diseño e implementación de ese sistema con el objetivo de lograr un proceso eficiente tanto en la gestión de la información como en la selección del personal para actividades docentes y productivas. Para llevar a cabo la creación de este sistema se tuvo como guía la metodología ágil XP, además de la utilización de PHP como lenguaje de programación, se puso en marcha un servidor apache y MySQL como gestor de base de datos. Como resultado se obtuvo un prototipo funcional del sistema que satisface las necesidades del cliente, además de la documentación pertinente obtenida a lo largo del trabajo.

Palabras Claves: *Software Educativo, Multimedia, Gestión.*

ÍNDICE

INTRODUCCIÓN 1

CAPÍTULO 1..... 4

1.1. INTRODUCCIÓN 4

1.2. SISTEMA DE GESTIÓN DE LA INFORMACIÓN 4

1.3. TENDENCIAS ACTUALES 4

1.4. SISTEMAS DE GESTIÓN DE INFORMACIÓN EXISTENTES NACIONALES E INTERNACIONALES 5

1.4.1. *Sistema de Gestión de Información sobre Cooperación Internacional (SIGCO)* 5

1.4.2. *Sistema de Gestión de Información del centro de estudios de medio ambiente y recursos naturales de La Universidad de Pinar del Río*..... 5

1.4.3. *Digital Learning SL*..... 6

1.4.4. *IDICT Villa Clara Servicio de Consultoría de Sistema de Gestión de Información*..... 6

1.5. TENDENCIAS Y TECNOLOGÍAS ACTUALES 7

1.5.1. *Arquitectura por capas* 7

1.5.2. *Lenguaje de Programación y Tecnologías del lado del Cliente* 9

1.5.3. *Lenguajes de Programación y Tecnologías del lado del Servidor*..... 10

1.6. SISTEMAS GESTORES DE BASES DE DATOS (SGBD) 13

1.7. METODOLOGÍA PARA EL DESARROLLO DEL SOFTWARE 15

1.8. SERVIDOR WEB 20

1.9. HERRAMIENTAS A UTILIZAR 21

1.10. CONCLUSIONES 22

CAPÍTULO 2..... 23

2.1 INTRODUCCIÓN..... 23

2.2 OBJETO DE ESTUDIO 23

2.2.1 *Procesos involucrados en el campo de acción* 23

2.2.2 *Objeto de Automatización* 23

2.2.3 *Propuesta de Sistema* 23

2.2.4 *Personas Relacionadas con la aplicación* 24

2.3 REQUISITOS NO FUNCIONALES DEL SISTEMA 25

2.4 CONCLUSIONES 26

CAPÍTULO 3..... 27

3.1 INTRODUCCIÓN.....	27
3.2 FASE EXPLORACIÓN.....	27
3.2.1 <i>Historias de usuario</i>	27
3.3 FASE PLANIFICACIÓN.....	30
3.3.1 <i>Estimación de esfuerzo por historia de usuario</i>	30
3.3.2 <i>Plan de iteraciones</i>	31
3.3.3 <i>Plan de duración de las iteraciones</i>	31
3.3.4 <i>Plan de entregas</i>	32
3.4 CONCLUSIONES.....	32
CAPÍTULO 4.....	33
4.1 INTRODUCCIÓN.....	33
4.2 DISEÑO DEL SISTEMA.....	33
4.2.1 <i>Tarjetas CRC</i>	34
4.3 DISEÑO DE LA BASE DE DATOS.....	39
4.4 FASE IMPLEMENTACIÓN.....	40
4.4.1 <i>Tareas de las historias de usuario desarrolladas en la iteración 1</i>	41
4.4.2 <i>Tareas de las historias de usuario desarrolladas en la iteración 2</i>	46
4.5 FASE PRUEBA.....	50
4.5.1 <i>Pruebas de aceptación</i>	50
4.5.2 <i>Pruebas de Aceptación de la Iteración 1</i>	51
4.5.3 <i>Pruebas de Aceptación de la Iteración 2</i>	58
4.6 CONCLUSIONES.....	83
CONCLUSIONES.....	84
RECOMENDACIONES.....	85
REFERENCIAS BIBLIOGRÁFICAS.....	86
BIBLIOGRAFÍA.....	88
ANEXOS.....	90
GLOSARIO DE TÉRMINOS.....	92

INTRODUCCIÓN

La Universidad de las Ciencias Informáticas fue creada como Universidad insigne de la producción de software en Cuba, desde sus inicios se pensó en varias estrategias de desarrollo rápido pero a la vez sólido, por lo que se hizo un estudio minucioso para el aprovechamiento óptimo de los recursos con que contaba el país. A esto se le han sumando nuevas tácticas de trabajo que han perfeccionado la calidad de las actividades realizadas. Sin duda, una de las estrategias, propias de la Universidad, que ha tenido un resultado apreciable es la conformación de los Polos Productivos.

El Polo de Software Educativo y Multimedia pretende brindar orientaciones específicas a las esferas que se tratan en el centro. Los miembros del Polo a través de la aplicación serán capaces de gestionar toda la información referente a cursos de pregrado, cursos de post grado, diplomado, maestría, eventos, tesis, investigaciones y proyecto según el permiso asignado. En el mismo con el incremento de los volúmenes de información se hace difícil un análisis integral por lo que el objetivo de la presente investigación es el desarrollo de un sistema para dar solución a este problema. Para esto se brinda una propuesta de aplicación Web con el objetivo de centralizar y facilitar el acceso a la información proveniente del Polo de Software Educativo y Multimedia, específicamente la que se confeccionan por cada área de trabajo.

En la actualidad se labora en 5 proyectos de exportación y 8 proyectos nacionales, se cuenta con 2 grupos internos, un grupo de gestión de medias que garantiza la obtención de los recursos mediáticos que se necesitan en los proyectos y otro de comercialización que se encarga del estudio de mercado, identificación de posibles clientes para los productos y de empresas a las que es posible aliarse para producir software. La vinculación a proyectos productivos es de 40 profesores y 165 estudiantes, de un total de 41 profesores y 345 estudiantes que son miembros del Polo. Básicamente se encuentra conformado por 5 áreas fundamentales: Formación, Investigación, Calidad, Arquitectura y Comercialización para su mejor control y organización. El Polo está siempre en espera de asumir nuevos proyectos lo que se traduce en el aumento considerable de actividades y por supuesto de responsabilidades.

Los volúmenes de información que se procesan crecen constantemente, por lo que se hace necesario un control más eficiente desde sus diferentes aristas tanto para la organización y el control de las actividades productivas, como la centralización de la

información de cada una de las áreas existentes dentro del Polo de Software Educativo y Multimedia. Aunque la información que se maneja en el Polo está sustentada en soporte digital, se dificulta el análisis integral de la misma, provocando un alto consumo de tiempo a las personas encargadas de la actividad y se hace demasiado complejo este proceso, además, no cumple con los requisitos de seguridad para manejar los datos generados dando lugar a una baja confiabilidad, dispersión y retraso en el flujo de información.

Después de realizar un análisis de la situación problemática, se define el siguiente **Problema a Resolver**:

¿Cómo facilitar la gestión de la información de cada una de las áreas existentes dentro del Polo de Software Educativo y Multimedia?

La gestión de la información es imprescindible para la toma de decisiones, por tanto el **Objeto de Estudio** está conformado por los *procesos de gestión y control de la información generada por las actividades de los Polos Productivos en la Universidad de las Ciencias Informáticas y el **Campo de Acción** se centra en las actividades del Polo de Software Educativo y Multimedia.*

El **Objetivo General** de este trabajo es *proveer a los profesores vinculados al Polo de Software Educativo y Multimedia de una aplicación que permita gestionar la información*, para ello los **Objetivos Específicos** trazados son:

1. *Estudiar los procesos de gestión de información que se desarrollan en el Polo de Software Educativo y Multimedia.*
2. *Estudiar el estado del arte.*
3. *Aplicar tecnologías, herramientas y metodologías para facilitar la gestión de información en el Polo.*

Para poder dar cumplimiento a estos objetivos de una forma completa y exitosa, se ha decidido desarrollar las siguientes **Tareas a cumplir por los estudiantes**:

1. *Estudio del estado actual de las áreas dentro del Polo e identificación de las relaciones entre ellas.*
2. *Selección de las tecnologías a utilizar.*
3. *Familiarización del equipo de desarrollo con las tecnologías y herramientas seleccionadas.*

4. *Realización de la implementación del sistema de forma iterativa obteniendo al final de cada iteración un prototipo funcional.*
5. *Realización de pruebas funcionales a la aplicación.*
6. *Conformación del documento de Tesis.*

Idea a Defender: *Si se desarrolla una aplicación que permita gestionar la información de las áreas del Polo de Software Educativo y Multimedia, se facilitará el manejo de los datos generados en este grupo.*

Posibles Resultados: *La herramienta con su correspondiente documentación centralizada.*

Capítulo 1

FUNDAMENTACIÓN TEÓRICA

1.1.INTRODUCCIÓN

En el presente capítulo se hace un análisis de cómo se encuentran a nivel internacional y nacional las tecnologías similares al sistema que se pretende desarrollar. Se lleva a cabo un estudio de las metodologías de desarrollo de software, así como de los servidores web, lenguajes de programación web y los gestores de base de datos; de ellos se analizan sus características, funciones y ventajas.

Además se exponen las propuestas de herramientas que posibiliten completar el producto requerido y finalmente se hace la selección de estas herramientas técnicas y metodologías para el desarrollo del sistema.

1.2.SISTEMA DE GESTIÓN DE LA INFORMACIÓN

Los sistemas de gestión de la Información para la gestión del conocimiento constituyen hoy una alternativa de imprescindible presencia en cada organización. Al permitir operar casi todos los activos tangibles e intangibles de la institución y llegar a convertirse en la herramienta integral de gerencia más cotizada y necesaria para alcanzar con éxito los resultados propuestos por la organización. ⁽¹⁾

Un sistema de gestión de la información es el conjunto de políticas y normas relacionadas entre sí que se establecen para el acceso y tratamiento de los recursos de información. Es un sistema diseñado para ayudar a los administradores a planificar y dirigir las operaciones de negocio y de organización. Un sistema de gestión de la información debe, por lo tanto, ayudar a los administradores del proyecto a saber qué información necesitan recabar, para tomar diferentes decisiones en distintos momentos.

1.3.TENDENCIAS ACTUALES

En los nuevos modelos de negocio la gestión de la información adquiere importancia estratégica. Las tendencias observadas en la práctica son: evolución hacia la denominada gestión de contenidos, que comprendería la gestión de documentos y

datos tanto internos como externos; aceptación definitiva de algunos documentos electrónicos en las organizaciones como forma válida de documento; necesidad creciente de gestionar electrónicamente información no estructurada en bases de datos; reconocimiento de la informática como una herramienta y no como base de la gestión de la información; cada vez menos importante la gestión de los soportes a favor de la accesibilidad de los contenidos; y, por último, previsión de la gestión de la información electrónica a medio-largo plazo. ⁽²⁾

1.4. SISTEMAS DE GESTIÓN DE INFORMACIÓN EXISTENTES NACIONALES E INTERNACIONALES

1.4.1. Sistema de Gestión de Información sobre Cooperación Internacional (SIGCO)

SIGCO es un sistema integral de gestión de la información para el manejo de programas desarrollados sobre la base del uso de mecanismos de Cooperación Internacional. Inicialmente ha sido concebido para el manejo de información sobre:

- Asistencia Humanitaria que reciben los países que enfrentan una situación de emergencia o que emprenden un plan de rehabilitación y/o reconstrucción.
- Las necesidades detectadas y diagnósticos de situación producidos por las autoridades nacionales en estos contextos.
- Los proyectos oficiales concebidos para aliviar las consecuencias de catástrofes naturales y/o sociales y avanzar hacia el desarrollo de planes de reconstrucción.

Sin embargo, en su implantación ha demostrado la flexibilidad de su estructura para ser utilizado aún en circunstancias de normal planificación de gestión. **SIGCO** ha sido desarrollado para disponer de información consolidada, detallada, validada, fácilmente accesible, especializada para grupos de usuarios, actualizada y versátil sobre la Cooperación Internacional. ⁽³⁾

1.4.2. Sistema de Gestión de Información del centro de estudios de medio ambiente y recursos naturales de La Universidad de Pinar del Río.

Ofrece una visión preliminar del diseño de un sistema de información para la gestión del conocimiento en la Universidad de Pinar del Río. Uno de los objetivos de este sistema es posibilitar un mejor desempeño informacional en la actividad investigativa del colectivo de investigadores de dicho centro de estudio. Para lo cual se realizó en

primera instancia un diagnóstico que reveló la situación problemática, la inexistencia de habilidades en la gestión de información en apoyo a la investigación y en segundo lugar la identificación de los principales flujos de información existentes dentro del grupo. Todo esto permitió orientar los principales objetivos del trabajo en la propuesta de diseño del Sistema de Gestión de Información para la Gestión del Conocimiento en el CEMARNA. ⁽¹⁾

1.4.3. Digital Learning SL

Digital Learning es una consultora de formación especializada en el área de las Tecnologías de la Información y la Comunicación. Su objetivo es diseñar acciones formativas para un amplio rango de alumnos, desde el universitario al directivo de empresa, con un denominador común: que puedan conocer las enormes posibilidades que Internet y las Tecnologías de la Información les ofrecen para mejorar en su desempeño personal, desarrollar su carrera profesional o facilitar la gestión de su entorno de trabajo. ⁽⁴⁾

1.4.4. IDICT Villa Clara Servicio de Consultoría de Sistema de Gestión de Información.

El servicio está basado en la necesidad de un reposicionamiento de la información como base del conocimiento, que constituye el eje central del cambio hacia una nueva sociedad con empresas perfeccionadas. Este aborda las acciones y beneficios del diseño e implementación del Sistema de Gestión de Información, que permiten diagnosticar y organizar los recursos de información, determinar el valor y disminuir los costos de la información. Además de evaluar la efectividad actual de la gestión de la información, se elaboran los mapas de flujos de la información por subsistemas y el general. Permite diseñar, organizar y formalizar la Gestión Documental y el trabajo de la Unidad de la Información, resultados que se perciben a través de la Intranet Corporativa de la organización. Propone un plan de medidas que permite a la empresa ascender a un estadio superior.

El proyecto se desarrolla a través de cuatro etapas:

- Diagnóstico preliminar
- Sistema de Gestión Documental
- Gestión de la Unidad de Información
- Arquitectura de contenidos (Intranet)

El sistema facilita a los clientes internos de la organización la información adecuada, en la forma correcta, para la persona indicada, al costo adecuado, en el tiempo oportuno, en el lugar apropiado, para tomar la acción correcta. ⁽⁵⁾

El estudio de estos Sistema de Gestión permitirá seleccionar algunas funcionalidades semejantes para gestionar la información así como incorporar otras específicas para el desarrollo de la aplicación con el objetivo de procesar la información referente a las distintas áreas existentes dentro del Polo de Software Educativo y Multimedia de forma más sencilla y eficiente cumpliendo con las expectativas del cliente.

1.5.TENDENCIAS Y TECNOLOGÍAS ACTUALES

1.5.1. ARQUITECTURA

Modelo-Vista-Controlador (MVC)

El patrón conocido como Modelo-Vista-Controlador (MVC) separa el modelado del dominio, la presentación y las acciones basadas en datos ingresados por el usuario en tres clases diferentes:

- Modelo. El modelo administra el comportamiento y los datos del dominio de aplicación, responde a requerimientos de información sobre su estado (usualmente formulados desde la vista) y responde a instrucciones de cambiar el estado (habitualmente desde el controlador).
- Vista. Maneja la visualización de la información.
- Controlador. Interpreta las acciones del ratón y el teclado, informando al modelo y/o a la vista para que cambien según resulte apropiado.

Tanto la vista como el controlador dependen del modelo, el cual no depende de las otras clases. Esta separación permite construir y probar el modelo independientemente de la representación visual. La separación entre vista y controlador puede ser secundaria en aplicaciones y, de hecho, muchos frameworks de interfaz implementan ambos roles en un solo objeto. En aplicaciones de Web, por otra parte, la separación entre la vista (el browser) y el controlador (los componentes del lado del servidor que manejan los requerimientos de HTTP) está mucho más taxativamente definida.

Arquitectura por capas

La programación por capas es un estilo de programación en la que el objetivo primordial es la separación entre de la lógica de negocios y la lógica de diseño. La ventaja principal de este estilo, es que el desarrollo se puede llevar a cabo en varios niveles y en caso de algún cambio sólo se ataca al nivel requerido sin tener que revisar entre código mezclado.

La arquitectura en tres capas cuenta con una interfaz gráfica que facilita al usuario el uso del sistema:

(Capa 1: Capa de Presentación) con una capa para centralizar la lógica de negocio, (Capa 2: Lógica de Negocio) y por último una capa que servirá para guardar los datos, (Capa 3: Base de Datos).

A continuación se muestra la figura que presenta la arquitectura en 3 capas.

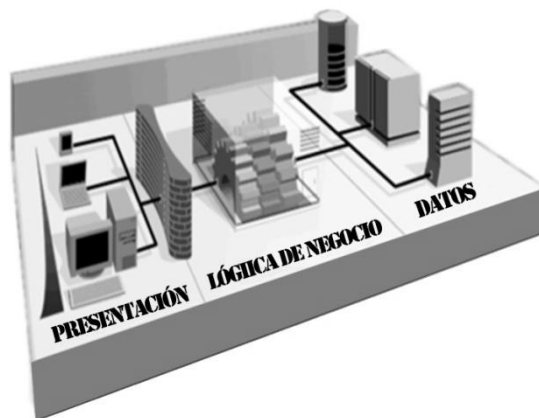


Figura 1 Arquitectura de 3 capas

Por lo antes descrito, más la experiencia que existe en los proyectos productivos de la Universidad y además la facilidad de integración y optimización de tiempo se utilizará el estilo n-capas específicamente el patrón de 3 capas.

Lenguaje de Programación

Un lenguaje de programación es un lenguaje diseñado para describir el conjunto de acciones consecutivas que un equipo debe ejecutar. Por lo tanto, un lenguaje de programación es un modo práctico para que los seres humanos puedan dar instrucciones a un equipo. ⁽⁶⁾

1.5.2. Lenguaje de Programación y Tecnologías del lado del Cliente.

- **Java Script**

Java Script es un lenguaje de programación interactivo del lado del cliente, se utiliza dentro del Hipertexto principalmente en páginas web. Es orientado a objetos y guiado por eventos ya que dispone de Herencia siguiendo el paradigma de programación basada en prototipos, de ahí deriva el dinamismo que pueden alcanzar las páginas que incluyan esta clase de código; dado a su compatibilidad con todos los navegadores modernos el código Java Script es interpretado dentro de las páginas web. Con Java Script es posible definir interactividad con el usuario de modo que el mayor recurso, y tal vez el único, con que cuenta este lenguaje es el propio navegador. Es un lenguaje de programación bastante sencillo y pensado para hacer las cosas con rapidez. Su característica principal, es la de ser un lenguaje de scripting, pero sobre todo, la de ser el lenguaje de scripting por excelencia y sin lugar a dudas, el más usado. Esta particularidad conlleva una notable serie de ventajas y desventajas, según el uso que se le deba dar y teniendo en cuenta la relación que se establece en el mecanismo cliente – servidor. ⁽⁷⁾

- **HTML (Hipertexto Markup Language)**

HTML, es un Lenguaje de marcas de Hipertexto (Hipertexto Markup Language), utilizado normalmente en la World Wide Web. Es muy sencillo y permite describir textos de forma estructurada y agradable, con enlaces, los cuales permiten conectar dos elementos entre sí o fuentes de información relacionadas. HTML es un sistema de etiquetas el cual está basado en especificar la estructura lógica del contenido (títulos, párrafos de texto normal, enumeraciones, definiciones, citas, entre otras) así como los diferentes efectos que se quieren dar (cursiva, negrita o un gráfico determinado), los mismos indican cómo debe verse la información. El entorno para trabajar es compatible con la mayoría de los sistemas operativos debido a que simplemente es un procesador de texto como el que ofrecen Windows (Bloc de notas), UNIX (el editor vi o ed). ⁽⁸⁾

- **CSS (Hojas de estilo en cascadas)**

Las hojas de estilo en cascada (Cascading Style Sheets, CSS) son un lenguaje formal usado para definir la presentación de un documento estructurado escrito en HTML o XML. Es el encargado de formular la especificación de las hojas de estilo que servirán de estándar para los agentes de usuarios o navegadores. CSS se utiliza para dar estilo a documentos HTML y XML, separando el contenido de la presentación. Los Estilos definen la forma de mostrar los elementos HTML y XML. CSS permite a los desarrolladores Web controlar el estilo y el formato de múltiples páginas Web al mismo tiempo. Cualquier cambio en el estilo marcado para un elemento en la CSS afectará a todas las páginas vinculadas a esa CSS en las que aparezca ese elemento.

1.5.3. Lenguajes de Programación y Tecnologías del lado del Servidor.

- **PHP (Personal Home Page)**

PHP (Personal Home Page) ha sido especialmente creado para el desarrollo de páginas web dinámicas y puede ser incluido con facilidad dentro del código HTML, con la posibilidad de usar programación de procedimientos o programación orientada a objetos. El nivel de familiarización con el lenguaje para el programador aprendiz es sorprendente, no obstante los profesionales se sentirán a gusto, pues PHP ofrece innumerables características de programación avanzada. Es gratuito, puede descargarse y utilizarse en cualquier aplicación personal o profesional de manera completamente libre y al ser de código abierto puede ser utilizado, modificado y redistribuido sin coste alguno.

PHP ha alcanzado gran popularidad y existe una amplia comunidad de desarrolladores y programadores que continuamente implementan mejoras en su código. Se caracteriza por una sencilla integración con múltiples Sistemas de Gestión de bases de datos, aunque MySQL es el que mejor trabaja con PHP, puede conectarse también con Postgre SQL, Oracle y cualquier otro Sistema de Gestión de bases de datos compatible. Además puede usarse en diversos sistemas operativos gratuitos de tipo UNIX, como Linux y FreeBSD, versiones comerciales de Unix, como Solaris e IRIX o en las diferentes versiones de Microsoft Windows. Está dotado de un gran número de funciones predefinidas que simplifican enormemente tareas habituales como descargar documentos, generar imágenes GIF, enviar correos electrónicos, trabajar con cookies y sesiones, establecer conexiones a otros servicios de red y generar documentos PDF.

- **ASP.NET**

ASP.NET es un conjunto de tecnologías de desarrollo de aplicaciones Web comercializado por Microsoft. Es usado por programadores para construir sitios Web domésticos, aplicaciones Web y servicios XML. Forma parte de la plataforma .NET y es la tecnología sucesora de Active Server Pages (ASP). ASP.NET es una nueva estructura de programación y fue diseñado con el objetivo de que dichas aplicaciones respondan rápidamente a las solicitudes de los usuarios, sin importar la cantidad de datos que se estén procesando en el servidor debido a que en una página ASP se pueden incluir: HTML plano, código de scripting y texto, no hay una distinción formal entre el contenido de una página y su comportamiento. ASP.NET impone un cierto orden sobre el modelo de programación estándar ASP, proporciona diversas mejoras en las cuales se destacan:

Rendimiento: se compila desde el código nativo, lo que permite mucho mejor rendimiento y un almacenamiento de la caché en el servidor.

Rapidez en programación: mediante diversos controles, se logra con unas pocas líneas y en menos de 5 minutos mostrar toda una base de datos y hacer rutinas complejas.

Servicios Web: trae herramientas para compartir datos e información entre distintos sitios.

Seguridad: tiene diversas herramientas que garantizan la seguridad de nuestras aplicaciones. ⁽⁹⁾

- **Java**

Java es un lenguaje de programación con el que se puede realizar cualquier tipo de programa. En la actualidad es un lenguaje muy extendido y cada vez cobra más importancia tanto en el ámbito de Internet como en la informática en general. Una de las principales características por las que Java se ha hecho muy famoso es que es un lenguaje independiente de la plataforma, esto en alguna medida es una ventaja significativa para los desarrolladores de software pues antes tenían que hacer un programa para cada sistema operativo, por ejemplo Windows, Linux y Apple. Esto lo consigue porque se ha creado una Máquina de Java para cada sistema que hace de puente entre el sistema operativo y el programa de Java y posibilita que este último se entienda perfectamente. Java está desarrollándose incluso para distintos tipos de

dispositivos además del ordenador como móviles, agendas y en general para cualquier cosa que se le ocurra a la industria. ⁽¹⁰⁾

¿Por qué utilizar PHP?

Luego de un análisis comparativo entre los diferentes lenguajes de programación más usados, se determina emplear PHP para el desarrollo de la aplicación debido a que, en ambientes Windows, PHP es mucho más estable que los demás lenguajes de programación, además de ser más rápido y utilizar menos recursos proporcionando numerosas ventajas:

1. PHP corre en cualquier plataforma utilizando el mismo código fuente.
2. La sintaxis de PHP es similar a la del lenguaje C, por lo que al tener experiencia en lenguajes del estilo C, se podrá entender rápidamente PHP, este es el caso de los desarrolladores del sistema.
3. Muchas interfaces distintas para cada tipo de servidor. PHP actualmente se puede ejecutar bajo Apache, IIS, AOLServer.
4. Puede conectarse con muchos motores de bases de datos tales como MS SQL, Oracle, Informix y PostgreSQL, destacando su conectividad con MySQL.
5. Con respecto a la rapidez, PHP generalmente es utilizado como módulo de Apache, lo que lo hace extremadamente veloz.
6. Es capaz de leer y manipular datos desde diversas fuentes, incluyendo datos que pueden ingresar los usuarios desde formularios HTML.
7. Tiene la capacidad de expandir su potencial utilizando su enorme cantidad de módulos.
8. Posee una amplia documentación en su Web oficial de Internet.
9. Pertenece a la alternativa de código abierto (Open Source).
10. Permite las técnicas de Programación Orientada a Objetos.
11. Permite crear los formularios para la web.
12. Cuenta con una biblioteca sumamente amplia para facilitar su funcionalidad.

PHP 5

En la implementación del sistema propuesto será utilizado PHP5. La selección se basa en que en esta versión, entre otras cosas, hace un cambio en el manejo de los objetos. Por ejemplo en PHP4 los objetos son tratados igual que otros tipos de datos básicos, como los enteros o los arreglos.

O sea, cuando se realizan operaciones sobre los objetos, como asignación de variables o cuando son pasados como parámetros a funciones, todo el objeto es copiado.

Por otra parte, en PHP5 todas las variables que nombran objetos son en realidad referencias. Otro de los elementos es que en PHP5 se incluyen modificadores de control de acceso para implementar el encapsulamiento, lo cual no existía en versiones anteriores. PHP5 tiene capacidad de conexión con la mayoría de los manejadores de base de datos que se utilizan en la actualidad.

Por todo lo anterior expuesto se decidió utilizar a PHP5 como el lenguaje encargado de modelar el sistema que se propone.

1.6.SISTEMAS GESTORES DE BASES DE DATOS (SGBD)

Los Sistemas Gestores de Bases de Datos (SGBD) son un tipo de software que sirven de interfaz entre la base de datos, el usuario y las aplicaciones que la utilizan y se compone de los siguientes elementos:

1. Un lenguaje de definición de datos.
2. Un lenguaje de manipulación de datos.
3. Un lenguaje de consulta.

Un Sistema Gestor de Base de Datos (SGBD) es un conjunto de programas que permiten crear y mantener una Base de datos, asegurando su integridad, confidencialidad y seguridad. Por tanto debe permitir:

1. Definir una Base de Datos: especificar tipos, estructuras y restricciones de datos.
2. Construir la Base de Datos: guardar los datos en algún medio controlado por el mismo SGBD.
3. Manipular los datos: realizar consultas, actualizarla, y generar informes.

• MySQL

MySQL es un sistema administrador de bases de datos relacionales (SGBDR), creado por la compañía MySQL AB fundada en 1995. Se trata de un SGBD multihilo y multiusuario, cuyo principal objetivo de diseño fue la velocidad (ofrece alto rendimiento debido a su rapidez de respuesta). Provee múltiples motores de almacenamiento, posibilita conexiones entre diferentes computadoras con distintos sistemas operativos y una integración perfecta con PHP. MySQL utiliza SQL (Structure Query Language),

el lenguaje estándar utilizado en el mundo para la consulta de bases de datos. Este gestor de bases de datos es, probablemente, el gestor más usado en el mundo del software libre, debido a su gran rapidez y facilidad de uso. Esta gran aceptación es debida, en parte, a que existen infinidad de librerías y otras herramientas que permiten su uso a través de gran cantidad de lenguajes de programación, además de su fácil instalación y configuración. Entre las características disponibles en las últimas versiones se pueden destacar las que se muestran a continuación: ⁽¹¹⁾

4. Amplio subconjunto del lenguaje SQL.
5. Disponibilidad en gran cantidad de plataformas y sistemas (Windows NT, 2000, 2003 y XP, Mac OS X, Linux, entre otros).
6. Varias opciones de almacenamiento, teniendo en cuenta si se desea velocidad en las operaciones o el mayor número de operaciones disponibles.
7. Transacciones y claves foráneas.
8. Conectividad segura.
9. Replicación.
10. Búsqueda e indexación de campos de texto. ⁽¹²⁾

- **Microsoft SQL Server**

Microsoft SQL Server es una plataforma de base de datos y análisis de datos que se utiliza en el procesamiento de transacciones en línea (OLTP) a gran escala, el almacenamiento de datos y las aplicaciones de comercio electrónico. SQL Server ofrece la tecnología y las funciones con las que pueden contar las organizaciones. Con avances significativos en áreas clave de la administración de datos empresariales, la productividad de los desarrolladores y la inteligencia empresarial, las ventajas de SQL Server son considerables y disímiles, entre las que se puede mencionar: ⁽¹³⁾

1. Aprovechamiento de los activos de datos.
2. Aumento de la productividad.
3. Reducción de la complejidad de la tecnología de la información.
4. Disminución del costo total de propiedad.
5. Microsoft SQL Server, al contrario de su más cercana competencia, no es multiplataforma, ya que sólo está disponible en Sistemas Operativos de Microsoft.

- **Oracle**

Oracle es un sistema de base de datos relacional extremadamente potente y flexible. Esta potencia y flexibilidad, sin embargo, implican también una cierta complejidad.

Para poder diseñar aplicaciones útiles basadas en Oracle es necesario entender como manipula Oracle así como los datos almacenados en el sistema. PL/SQL es una herramienta de gran importancia diseñada para la manipulación de datos, tanto internamente dentro de Oracle como externamente, en las propias aplicaciones. PL/SQL está disponible en diversos entornos, cada uno de los cuales tiene diferentes ventajas. Es una aplicación propietaria y sus precios son muy altos en el mercado, ascienden a los miles de euros, actualmente se encuentra en la versión 9. ⁽¹⁴⁾

¿Por qué utilizar MySQL?

Un buen punto es debido a los costos, otro punto a favor es el gran soporte que presenta además de la gran diversidad de clientes para hacer más interactiva la administración del servidor. Por otra parte, en aplicaciones web su acceso es más rápido comparado con otros motores de bases de datos. Permite el acceso a las bases de datos de forma simultánea por varios usuarios y aplicaciones, seguridad, en forma de permisos y privilegios, determinados para los usuarios. Es un sistema gestor de base de datos muy potente para consulta y estandarizado, de modo que las consultas hechas usando SQL son fácilmente portables a otros sistemas o plataformas.

En concreto usar **MySQL** tiene ventajas adicionales:

1. Posibilita manipular bases de datos enormes.
2. Permite conexiones entre diferentes máquinas con distintos sistemas operativos.
3. Es multihilo con lo que puede beneficiarse de sistemas multiprocesadores.
4. Permite manejar multitud de tipos para columnas.
5. Permite manejar registros de longitud fija o variable. ⁽¹⁵⁾

1.7. METODOLOGÍA PARA EL DESARROLLO DEL SOFTWARE

Muchas veces el proceso de desarrollo de software resulta riesgoso y se convierte en una tarea difícil hallar el modo de controlar su curso de principio a fin. El problema principal radica en cómo coordinar todas las actividades que comprende el desarrollo de un proyecto de software, sobre todo si se trata de un proyecto de gran envergadura. De modo que se torna imprescindible contar con una forma organizada y adecuadamente estructurada para trabajar. Se necesita un proceso que integre las múltiples fases del desarrollo, un método común, un proceso que. ⁽¹⁶⁾

1. Proporcione una guía para ordenar las actividades de un equipo.
2. Dirija las tareas de cada desarrollador por separado y del equipo como un todo.
3. Especifique los artefactos que deben desarrollarse.
4. Ofrezca criterios para el control y la medición de los productos y actividades del proyecto.

- **Rational Unified Process (RUP)**

El Rational Unified Process (Proceso Unificado de Rational) es un proceso de desarrollo de software, cuyos modelos y artefactos se expresan en el Lenguaje Unificado de Modelado. RUP es *una metodología robusta* que puede ser adaptado a proyectos de mayor o menor complejidad, aplicable a diferentes esferas y ajustable a las necesidades de cada organización. Se trata de un proceso iterativo e incremental debido a que se basa en la evolución de prototipos ejecutables que se muestran a los usuarios y clientes.

Se caracteriza por ser centrado en la arquitectura porque establece refinamientos sucesivos de una arquitectura ejecutable, construida como un modelo evolutivo de manera que no se afecte de forma significativa ante posibles modificaciones, para lograr finalmente una arquitectura comprensible, adaptable y robusta. Por último, RUP está dirigido por los casos de uso, pues guía el desarrollo del proyecto manteniendo como un aspecto de vital importancia la satisfacción del usuario y no sólo teniendo en cuenta las funcionalidades del sistema sino que permite controlar el proceso de desarrollo del proyecto al mismo tiempo que es elaborado, quedando conformada, a su vez, una guía para posteriores mejoras del producto.

En RUP se han agrupado las actividades en grupos lógicos definiéndose 9 flujos de trabajo principales. Los 6 primeros son conocidos como flujos de ingeniería y los tres últimos como de apoyo.

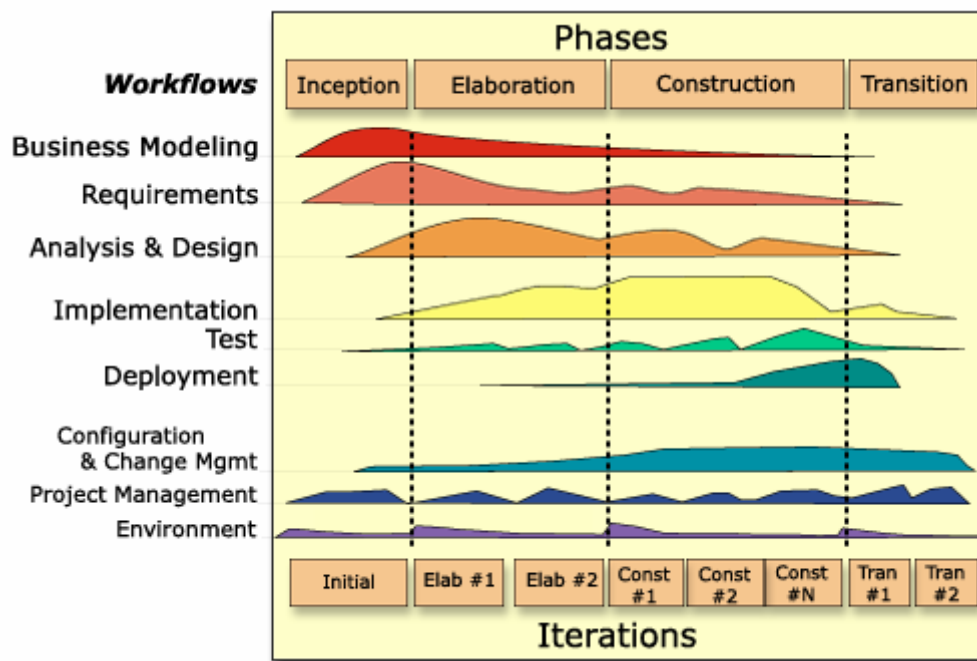


Figura 2 Flujo de trabajo

1. Flujos de trabajo:

- **Modelamiento del negocio:** Describe los procesos de negocio, identificando quiénes participan y las actividades que requieren automatización.
- **Requerimientos:** Define qué es lo que el sistema debe hacer, para lo cual se identifican las funcionalidades requeridas y las restricciones que se imponen.
- **Análisis y diseño:** Describe cómo el sistema será realizado a partir de la funcionalidad prevista y las restricciones impuestas (requerimientos), por lo que indica con precisión lo que se debe programar.
- **Implementación:** Define cómo se organizan las clases y objetos en componentes, cuáles nodos se utilizarán, la ubicación en ellos de los componentes y la estructura de capas de la aplicación.
- **Prueba (Testeo):** Busca los defectos a lo largo del ciclo de vida.
- **Instalación:** Produce release del producto y realiza actividades (empaquete, instalación y asistencia a usuarios) para entregar el software a los usuarios finales.
- **Administración del proyecto:** Involucra actividades con las que se busca producir un producto que satisfaga las necesidades de los clientes.

- **Administración de configuración y cambios:** Describe cómo controlar los elementos producidos por todos los integrantes del equipo de proyecto en cuanto a: utilización/actualización concurrente de elementos, control de versiones, entre otros.
- **Ambiente:** Contiene actividades que describen los procesos y herramientas que soportarán el equipo de trabajo del proyecto; así como el procedimiento para implementar el proceso en una organización.

2. Fases

- **Conceptualización (Concepción o Inicio):** Se describe el negocio y se delimita el proyecto describiendo sus alcances con la identificación de los casos de uso del sistema.
 - **Elaboración:** Se define la arquitectura del sistema y se obtiene una aplicación ejecutable que responde a los casos de uso que la comprometen. A pesar de que se desarrolla a profundidad una parte del sistema, las decisiones sobre la arquitectura se hacen sobre la base de la comprensión del sistema completo y los requerimientos (funcionales y no funcionales) identificados de acuerdo con el alcance definido.
 - **Construcción:** Se obtiene un producto listo para su utilización que está documentado y tiene un manual de usuario. Se obtiene uno o varios release del producto que han pasado las pruebas. Se ponen estos release a consideración de un subconjunto de usuarios.
 - **Transición:** El release ya está listo para su instalación en las condiciones reales. Puede implicar reparación de errores. ⁽¹⁶⁾
- **Extreme Programming (XP)**

Es una de las metodologías de desarrollo de software más exitosas en la actualidad utilizada para proyectos de corto plazo y equipo de desarrollo pequeño. Esta es una metodología ágil centrada en potenciar las relaciones interpersonales como clave para el éxito en el desarrollo de software, promoviendo el trabajo en equipo, preocupándose por el aprendizaje de los desarrolladores, y propiciando un buen clima de trabajo. XP se basa en realimentación continua entre el cliente y el equipo de desarrollo, comunicación fluida entre todos los participantes, simplicidad en las soluciones implementadas y coraje para enfrentar los cambios. XP se define especialmente para proyectos con requisitos imprecisos y muy cambiantes, y donde existe un alto riesgo

técnico. La metodología consiste en una programación rápida o extrema, cuya particularidad es tener como parte del equipo al usuario final, pues es uno de los requisitos para llegar al éxito del proyecto. ⁽¹⁷⁾

Principalmente esta metodología se caracteriza por:

1. Pruebas Unitarias: Se basa en las pruebas realizadas a los principales procesos, de tal manera que se adelanta en algo hacia el futuro, haciendo pruebas de las fallas que pudieran ocurrir. Es como adelantarse a obtener los posibles errores.
2. Refabricación: Se basa en la reutilización de código, para lo cual se crean patrones o modelos estándares, siendo más flexible al cambio.
3. Programación en pares: Una particularidad de esta metodología es que propone la programación en pares, la cual consiste en que dos desarrolladores participen de un proyecto en una misma estación de trabajo. Cada miembro lleva a cabo la acción que el otro no está haciendo en ese momento. Es como el chofer y el copiloto: mientras uno conduce, el otro consulta el mapa.

Lo fundamental en este tipo de metodología es:

1. La comunicación, entre los usuarios y los desarrolladores.
2. La simplicidad, al desarrollar y codificar los módulos del sistema.

La retroalimentación, concreta y frecuente del equipo de desarrollo, el cliente y los usuarios finales para así impedir la aparición de riesgos críticos que perjudiquen la realización con éxito del proyecto y simultáneamente, ayuda en gran medida a disminuir el tiempo de desarrollo y sus costos.

¿Por qué usar Extreme Programming XP?

Extreme Programming es una de las variantes de las metodologías ágiles con más aceptación en la comunidad de desarrollo, se programa de acuerdo con los deseos del cliente. El equipo de trabajo lo conforman el jefe de Polo y los desarrolladores del sistema, se rige por valores y principios tales como: comunicación, simplicidad, retroalimentación y coraje. Los desarrolladores tienen la posibilidad de programar más organizado, esto da la medida de una menor tasa de errores, el cliente tiene un mejor control de las prioridades y se realizan pruebas continuas durante el desarrollo de la aplicación. Una de las herramientas más importantes de la XP es el desarrollo orientado a pruebas, ya que utiliza las pruebas unitarias como eje de todo desarrollo.

⁽¹⁸⁾

1.8.SERVIDOR WEB

Un servidor web es un programa que se encarga de mantenerse a la espera de peticiones HTTP llevadas a cabo por un cliente HTTP que suele conocerse como navegador. El navegador realiza una petición al servidor y éste le responde con el contenido que el cliente solicita. El servidor responde al cliente enviando el código HTML de la página, una vez recibido el código, lo interpreta y lo muestra en pantalla. El cliente es el encargado de interpretar el código HTML, es decir, de mostrar las fuentes, los colores y la disposición de los textos y objetos de la página mientras que el servidor tan sólo se limita a transferir el código de la página sin llevar a cabo ninguna interpretación de la misma. ⁽¹⁹⁾

- **Apache**

Apache es considerado el más utilizado de los servidores web de todo el mundo. Este es un software libre que se desarrolla dentro del proyecto HTTP Server de la Apache Software Foundation. Es un servidor altamente configurable de diseño modular. Una de las características más importantes de Apache, es que se utiliza en casi todas las plataformas. Al principio solo funcionaba con los servidores web UNIX, pero en la actualidad no solo funciona con todas las versiones de UNIX sino que también lo hace con Windows 95, 98, ME, 2000, XP y Server 2003. Por otra parte, Apache permite la indexación de directorios, uso de sobrenombres con las carpetas, negociación de contenidos, informes configurables sobre los errores HTTP, programas CGI, administración de recursos para los procesos emparentados, mapas de imágenes para los servidores, reescritura, corrección de URL y provee interfaz con todos los sistemas de autenticación. ⁽²⁰⁾

- **Internet Information Servers (IIS)**

Es la solución de Microsoft a las necesidades de las empresas y usuarios de enviar y recibir la información no sólo de sus clientes si no también en el entorno de la empresa. IIS engloba una serie de herramientas administrativas que le permitirán controlar sitios Web, FTP, SMTP (correo saliente) y Servicio de noticias. Dispone también del soporte necesario para crear páginas dinámicas (ASP), lenguaje de aplicaciones para Internet bastante extendido permitiendo la conexión y acceso a bases de datos, consiguiendo realizar aplicaciones Web dinámicas y escalables. ⁽²¹⁾

1.9.HERRAMIENTAS A UTILIZAR

- **Dreamweaver**

Es uno de los programas más utilizado en el sector del diseño y la programación Web, por sus funcionalidades, su integración con otras herramientas como Adobe Flash y recientemente, por su soporte de los estándares del World Wide Web Consortium. Su principal competidor es Microsoft FrontPage. Tiene soporte tanto para edición de imágenes como para animación a través de su integración con otras herramientas. Dreamweaver permite al usuario utilizar la mayoría de los navegadores Web instalados en su ordenador para previsualizar las páginas Web. También dispone de herramientas de administración de sitios dirigidas a principiantes como, por ejemplo, la habilidad de encontrar y reemplazar líneas de texto y código por cualquier tipo de parámetro especificado, hasta el sitio Web completo. El panel de comportamientos también permite crear Java Script básico sin conocimientos de código. Permite la conexión a Bases de Datos como MySQL y Microsoft Access, para filtrar y mostrar el contenido utilizando tecnología de script como, por ejemplo, ASP (Active Server Pages), ASP.NET, ColdFusion, JSP (JavaServer Pages), PHP, sin necesidad de tener experiencia previa en programación. ⁽²²⁾

- **Cliente de Base Datos EMS SQL Manager 2005**

Se utilizará para la manipulación de la base de datos el EMS SQL Manager 2005 Lite para MySQL versión 3.7.2.5. Se decide emplear esta herramienta por las facilidades que brinda para la administración, desarrollo y mantenimiento de BD. Su interfaz gráfica es muy intuitiva e incluye un modo guiado de trabajo. Incluye además un editor de sentencias SQL, que apoya en la realización de procedimientos almacenados y consultas complejas que aceleran las búsquedas y recuperaciones de información útil en las aplicaciones. ⁽²³⁾

- **Zend Studio**

Zend Studio es uno de los ambientes de desarrollo integrado o Integrated Development Environment (IDE) disponible para desarrolladores profesionales que agrupa todos los componentes de desarrollo necesarios para el ciclo de desarrollo de aplicaciones PHP. Zend Studio acelera los ciclos de desarrollo y simplifica los proyectos complejos. Proporciona una serie de ayudas que pasan desde la creación y gestión de proyectos hasta la depuración de código. Consta de dos partes en las que se dividen las funcionalidades de parte del cliente y las del servidor. La del cliente

contiene el interfaz de edición y la ayuda. Permite además hacer depuraciones simples de scripts, aunque para disfrutar de toda la potencia de la herramienta de depuración habrá que disponer de la parte del servidor, que instala Apache y el módulo PHP, o en caso de que estén instalados, los configura para trabajar juntos en depuración. ⁽²⁴⁾

1.10. CONCLUSIONES

En este capítulo se fundamentaron las tecnologías más apropiadas para el desarrollo de la aplicación, donde se propone, teniendo en cuenta las tendencias actuales, la construcción de la base de datos utilizando el sistema gestor MySQL, la programación por parte del servidor se hará con PHP 5.0 y por parte del cliente se utilizará Java Script , HTML y CCS, se utilizará la metodología Extreme Programming XP, que a su vez, para el modelado físico de la base de datos se utiliza la herramienta Cliente de Base Datos EMS SQL Manager 2005. El avance de los medios enfocados a la Web se encuentra en un desarrollo vertiginoso y acelerado lo que permitió que se haya podido seleccionar de una gran gama de productos los más adaptables al sistema que se desea desarrollar.

Capítulo 2

CARACTERÍSTICAS DEL SISTEMA

2.1 INTRODUCCIÓN

En el presente capítulo se aborda sobre las principales características del sistema, se describe el objeto de estudio en el cual se llevan a cabo los procesos involucrados en el campo de acción, objetos de automatización y el desarrollo de la propuesta de solución, se especifican las personas relacionadas con la aplicación así como se exponen los requisitos no funcionales para el funcionamiento del sistema.

2.2 OBJETO DE ESTUDIO

2.2.1 Procesos involucrados en el campo de acción

El control así como el buen manejo del flujo de información generado por las diferentes áreas del polo de Software Educativo y Multimedia de la Facultad 8 se hace imprescindible a la hora de organizar todo tipo de evento o actividad ya sea producir o superarse potencialmente en materia informática, por ello se requiere una completa retroalimentación y gestión de los procesos y de esta forma brindar un mejor funcionamiento en el Polo Productivo. Actualmente para el mejor desempeño de las acciones se conformaron cinco esferas (Formación, Investigación, Calidad, Arquitectura y Comercialización) siendo las dos primeras las más significativas y con mayor situación crítica en lo que confiere a la elaboración, mantenimiento y flujo de documentos informáticos necesarios para el trabajo de los desarrolladores.

2.2.2 Objeto de Automatización

Se quiere automatizar el control y flujo de la información de cada área del Polo manteniendo una estrecha relación entre las mismas. Esta centralización permitirá el fácil acceso, subida, descarga y actualización de los documentos e información en los que trabaja cada uno de los grupos.

2.2.3 Propuesta de Sistema

Se ha decidido darle solución al problema mediante una aplicación Web que facilite la centralización, manejo y control de la documentación del Polo con la menor cantidad de contratiempos. Para ello se ha dividido en tres módulos:

1. Proyecto
2. Formación
3. Investigación

Estos tres módulos serán los encargados de la centralización de la información generadas por el Polo, estos permitirán:

1. Gestionar toda la información generada por las actividades que solo podrán ser realizadas por los jefes de áreas en su correspondiente área y el administrador del sistema.
2. Navegar y consultar el sistema, lo cual podrá ser efectuado por cualquier usuario del sistema ya sea invitado, miembro, jefe de área o jefe del Polo.

2.2.4 Personas Relacionadas con la aplicación

Se denomina persona relacionada con el sistema a aquella que interactúa e intercambia con el sistema y obtiene resultados de los procesos desarrollados en la aplicación. Además aquella que interactúa con la misma sin poder hacer uso de las secciones privilegiadas del sistema.

Personas relacionadas con el sistema	Justificación
Asesores	Es la persona encargada de atender y administrar el área correspondiente, actualiza e intercambia con la parte del sistema a la que tiene acceso pleno.
Jefe del Polo	Es la persona encargada de gestionar los miembros del Polo así como atender su área correspondiente, actualizando e intercambiando información con el sistema.

La interfaz de usuario: La aplicación presenta una interfaz sencilla de manejar y agradable que permite el acceso a cualquier parte de la misma o a alguna funcionalidad con no más de dos clic. El objetivo principal de la interfaz es manipular la información, ya sea accediendo a la documentación puntualizada o actualizando los archivos anteriormente fijados.

2.3 REQUISITOS NO FUNCIONALES DEL SISTEMA

- **Requerimientos de apariencia o interfaz externa**

La aplicación propuesta será usada por personas que tengan un conocimientos medio de informática, por lo que la interfaz debe ser amigable y fácil de usar, de manera que no sea una dificultad para los usuarios el trabajo con la misma en especial para aquellos que poseen pocas habilidades en el manejo de aplicaciones Web. La comunicación entre el servidor de base de datos y el servidor Web será mediante el protocolo *TCP/IP*, entre las máquinas clientes y el servidor Web será por HTTP y entre el servidor y el directorio activo mediante el protocolo de red *LDAP* (Lightweight Directory Access Protocol).

- **Requerimientos de usabilidad**

A los administradores finales de la aplicación se les dará un adiestramiento básico en el uso de la aplicación. Estas personas tendrán un nivel de acceso amplio en la aplicación para que puedan dar respuesta a cada incidente ocurrido.

- **Requerimientos de rendimiento**

Para un funcionamiento óptimo de la aplicación se seguirán las diferentes técnicas de elaboración de la Web, que faciliten el rápido acceso a sus páginas. La eficiencia del producto estará determinada en gran medida por el aprovechamiento de los recursos que se disponen en el modelo cliente/servidor, y la velocidad de la consultas de la base de datos. La aplicación propuesta debe ser rápida y tiempo de respuesta debe ser el mínimo posible, adecuado a la rapidez con que el cliente requiere la respuesta a su petición.

- **Requerimiento de portabilidad**

Las herramientas podrán ser usadas bajo cualquier sistema operativo de Windows NT en adelante o cualquier distribución de Linux. El servidor Web y el servidor de Base de Datos pueden estar en la misma PC sin ocasionar problema.

- **Requerimientos de seguridad**

Confiabilidad: La información manejada por el sistema debe estar protegida de acceso no autorizado.

Integridad: La información manejada por el sistema debe ser objeto de cuidadosa protección contra la corrupción y estados de inconsistencia.

Disponibilidad: La aplicación deberá estar disponible en todo momento para aquellas personas con acceso a la información y los mecanismos utilizados para lograr la seguridad no deben ser un obstáculo a los usuarios para obtener los datos deseados en un momento dado.

- **Requerimientos de software**

En las computadoras de los usuarios solo se requiere un navegador Web, bajo cualquier sistema operativo de Windows NT en adelante o cualquier distribución de Linux. En el servidor de base de datos se requiere de Windows NT en adelante o cualquier distribución de Linux y como gestor de base de datos MySQL.

- **Requerimientos de hardware**

En el cliente se requiere una computadora con 128 MB de RAM como mínimo, el servidor Web junto con el servidor de base de datos debe tener 256 MB de RAM y 20GB de disco duro mínimo y todas las PCs implicadas en el funcionamiento de la aplicación deben estar conectadas a la red.

- **Restricción en el diseño y la implementación**

Se debe realizar una aplicación que permita dar respuesta en el menor tiempo posible, garantizando la calidad del sistema. Para garantizar el desarrollo de la aplicación se utilizará la metodología XP.

2.4 CONCLUSIONES

En el capítulo se han abordado los principales problemas que existen en el Polo de Software Educativo y Multimedia, demostrándose la necesidad de crear una herramienta que opere y controle toda la información generada. Se presentó una breve descripción de la propuesta del sistema y las actividades fundamentales que debe implementar. Finalmente se definen los requisitos no funcionales del sistema para el obtener los resultados más apropiados.

Capítulo 3

EXPLORACIÓN Y PLANIFICACIÓN

3.1 INTRODUCCIÓN

En el presente capítulo se abordan las fases de exploración y planificación pertenecientes a la metodología de desarrollo XP (Extreme Programming) utilizada para la implementación del sistema que se propone. Además se exponen los artefactos generados durante el transcurso de dichas fases.

3.2 FASE EXPLORACIÓN

La metodología de desarrollo XP comienza con la fase de exploración. Durante esta se realiza el proceso de identificación de las historias de usuario, así como la familiarización de los equipos de trabajo con las tecnologías y herramientas seleccionadas para la construcción del proyecto.

3.2.1 Historias de usuario

Las historias de usuario son la forma en que se especifican en XP (Extreme Programming) los requisitos del sistema. Estas se escriben desde la perspectiva del cliente aunque los desarrolladores pueden brindar también su ayuda en la identificación de las mismas. El contenido de estas debe ser concreto y sencillo. Durante la fase de exploración se identificaron ocho historias de usuario, las cuales se detallan a continuación:

HU: Gestionar miembro

Historia de Usuario	
Número: 1	Nombre historia: Gestionar miembro
Usuario: Jefe del Polo, Asesores	
Prioridad en negocio: Alta	Riesgo en desarrollo: Alto
Puntos estimados: 1	Iteración asignada: 1

Programador responsable: Yan Monteagudo y Ernesto Muñoz
Descripción: Se le brinda la posibilidad al Jefe Polo a que realice las acciones de insertar y eliminar miembros al sistema, además de editar y mostrar los datos de cada miembro del Polo y a su vez los Asesores de cada área tienen la posibilidad de editar los datos pertenecientes a su área y mostrar los datos almacenados.
Observaciones: Si el usuario ya existe no podrá insertarse.

HU: Autenticar

Historia de Usuario	
Número: 2	Nombre historia: Autenticar
Usuario: Jefe del Polo, Asesores	
Prioridad en negocio: Alta	Riesgo en desarrollo: Medio
Puntos estimados: 0.5	Iteración asignada: 1
Programador responsable: Yan Monteagudo y Ernesto Muñoz	
Descripción: Se brinda la posibilidad de que el Jefe Polo y los Asesores accedan al sistema introduciendo sus datos (usuario y contraseña) con la finalidad de verificar y otorgarle los permisos según el rol que cumpla dentro de la aplicación.	
Observaciones: El usuario debe ser válido.	

HU: Publicar

Historia de Usuario	
Número: 3	Nombre historia: Publicar
Usuario: Jefe del Polo, Asesores	
Prioridad en negocio: Alta	Riesgo en desarrollo: Medio
Puntos estimados: 0.5	Iteración asignada: 1
Programador responsable: Yan Monteagudo y Ernesto Muñoz	
Descripción: Se brinda la posibilidad tanto al Jefe Polo como a los Asesores de	

publicar información de cursos, diplomados, maestrías, eventos, tesis, investigaciones y proyectos según los permisos que se le asignen.

Observaciones: -

HU: Gestionar Actividades de Formación

Historia de Usuario	
Número: 4	Nombre historia: Gestionar Actividades de Formación
Usuario: Asesores	
Prioridad en negocio: Alta	Riesgo en desarrollo: Alto
Puntos estimados: 1	Iteración asignada: 2
Programador responsable: Yan Monteagudo y Ernesto Muñoz	
Descripción: Se realizan las acciones de crear, editar y eliminar tanto cursos como diplomados y maestrías. El Asesor del área de Formación tiene la posibilidad de gestionar, actualizar y obtener reportes de las diferentes actividades de formación creadas para la superación de los miembros.	
Observaciones: -	

HU: Gestionar Actividades de Investigación

Historia de Usuario	
Número: 5	Nombre historia: Gestionar Actividades de Investigación
Usuario: Asesores	
Prioridad en negocio: Alta	Riesgo en desarrollo: Alto
Puntos estimados: 1	Iteración asignada: 2
Programador responsable: Yan Monteagudo y Ernesto Muñoz	
Descripción: Se realizan las acciones de crear, editar y eliminar tanto eventos como tesis e investigaciones. El Asesor del área de Investigaciones tiene la	

posibilidad de gestionar, actualizar y obtener reportes de las diferentes actividades de investigación creadas para la superación de los miembros.

Observaciones: -

HU: Gestionar Proyectos

Historia de Usuario	
Número: 6	Nombre historia: Gestionar Proyectos
Usuario: Jefe del Polo	
Prioridad en negocio: Alta	Riesgo en desarrollo: Alto
Puntos estimados: 1	Iteración asignada: 2
Programador responsable: Yan Monteagudo y Ernesto Muñoz	
Descripción: Se realizan las acciones de crear y editar proyectos. El Jefe de Polo tiene la posibilidad de gestionar, actualizar y obtener reportes de los diferentes proyectos existentes en el Polo de Software Educativo y Multimedia.	
Observaciones: Los proyectos no se deben eliminar aunque ya esté terminado, estos datos se deben quedar guardados para un futuro.	

3.3 FASE PLANIFICACIÓN

Durante la fase de planificación se realiza una estimación del esfuerzo que costará implementar cada historia de usuario. Este se expresa utilizando como medida el punto. Un punto se considera como una semana ideal de trabajo donde los miembros de los equipos de desarrollo trabajan el tiempo planeado sin ningún tipo de interrupción. Esta estimación incluye todo el esfuerzo asociado a la implementación de la historia de usuario, por ejemplo: las pruebas unitarias, la integración y refactorización del código y la preparación y ejecución de las pruebas de aceptación.

3.3.1 Estimación de esfuerzo por historia de usuario

Para el desarrollo del sistema propuesto en este trabajo se realizó una estimación del esfuerzo para cada una de las historias de usuario identificadas, llegándose a los resultados que se muestran en la siguiente tabla.

Estimación de esfuerzo por historia de usuario

Historia de Usuario	Puntos estimados
Gestionar miembro	1
Autenticar	0.5
Publicar	0.5
Gestionar Actividades de Formación	1
Gestionar Actividades de Investigación	1
Gestionar Proyectos	1

3.3.2 Plan de iteraciones

Una se vez identificadas las historias de usuario del sistema y estimado el esfuerzo dedicado a la realización de cada una de estas se procede a la planificación de la etapa de implementación del proyecto. De acuerdo a lo mencionado anteriormente se decidió realizar esta en dos iteraciones, detalladas a continuación.

Iteración 1

En esta iteración se realizarán las historias de usuarios que van a dar una idea de cómo quedará la aplicación aunque todavía estará en sus inicios.

Iteración 2

La implementación de las historias de usuarios en esta iteración proporcionará una idea completa de la aplicación la cual al terminar la implementación quedará terminado el sistema.

3.3.3 Plan de duración de las iteraciones

Como parte del ciclo de vida de un proyecto utilizando XP se crea el plan de duración de cada una de las iteraciones, en este caso se hace para el único equipo de desarrollo con que se cuenta. Este plan se encarga de mostrar las historias de usuario que serán abordadas en cada una de las iteraciones, así como la duración estimada de estas últimas y el orden en que se implementarán.

Plan de duración de las iteraciones

Iteración	Orden de las historias de usuarios	Duración de las iteraciones
Iteración 1	Gestionar miembro Autenticar Publicar	2 semanas
Iteración 2	Gestionar Actividades de Formación.	3 semanas

	Gestionar Actividades de Investigación. Gestionar Proyecto.	
--	--	--

3.3.4 Plan de entregas

A continuación se presenta el plan de entregas ideado para la fase de implementación. Como producto del mismo se harán *releases* del sistema al finalizar cada iteración en la fecha aproximada que se indica en la siguiente tabla.

Plan de entregas

Sistema	Final 1ra Iteración 4ta semana de Abril.	Final 2da Iteración 3ra semana de mayo
Sistema de información del Polo de Software Educativo y Multimedia	0.1	0.2

3.4 CONCLUSIONES

En el presente capítulo se abordó todo lo referente al desarrollo del sistema basado en la metodología XP, identificando seis historias de usuario haciendo una descripción detallada del negocio de las mismas en la fase de exploración, y en la fase de planificación se generaron los artefactos de estimación de esfuerzo que costara implementar cada historia de usuario, el plan de duración de las iteraciones para hacer entrega del producto al final de las mismas en el tiempo requerido y el plan de entrega del sistema basado en las fechas de las iteraciones anteriores.

Capítulo 4

IMPLEMENTACIÓN Y PRUEBA

4.1 INTRODUCCIÓN

La Metodología XP plantea que la implementación de un software debe realizarse de forma iterativa, obteniendo al culminar cada iteración un producto funcional que debe ser probado y mostrado al cliente para incrementar la visión de los desarrolladores con la opinión de éste. En el presente capítulo se detallan las dos iteraciones llevadas a cabo durante la etapa de construcción del sistema, exponiéndose las tareas generadas por cada historia de usuario, así como las pruebas de aceptación efectuadas sobre el sistema.

4.2 DISEÑO DEL SISTEMA

Para el diseño de las aplicaciones, la metodología XP no requiere la presentación del sistema mediante diagramas de clases utilizando notación UML, en su lugar se usan otras técnicas como las tarjetas CRC (Contenido, Responsabilidad y Colaboración). No obstante el uso de estos diagramas puede aplicarse siempre y cuando influyan en el mejoramiento de la comunicación, no sea un peso su mantenimiento, no sean extensos y se enfoquen en la información importante.

Sub-módulos a desarrollar para la aplicación

1. Gestionar Miembro.
2. Gestionar Curso.
3. Gestionar Superación.
4. Gestionar Evento.
5. Gestionar Tesis.
6. Gestionar Investigaciones.
7. Gestionar Proyecto.

4.2.1 Tarjetas CRC

Con el objetivo de hacer entendible las funcionalidades encapsuladas en estos módulos, se definen una tarjeta CRC por cada uno, con la finalidad de obtener un diseño simple y no incurrir en la implementación de características que no son necesarias.

Miembro polo	
Funcionalidades	Colaboraciones
Insertar miembro	Pre-grado
Eliminar miembro	Post-grado
Editar miembro	Superación
Devolver miembro	Evento
Devolver todos los cursos en que participó un estudiante en un tiempo determinado	Investigación
Devolver las actividades de formación en que ha participa un profesor.	
Devolver las actividades de formación en que está matriculado un miembro.	
Devolver los cursos en los que un miembro ha participado como profesor.	
Devolver los eventos en que ha participado un miembro en un período determinado.	
Devolver las publicaciones que ha hecho un miembro en un período determinado.	

Estudiante	
Funcionalidades	Colaboraciones
Insertar estudiante	Miembro_polo
Eliminar estudiante	
Editar estudiante	
Devolver estudiante	

Profesor	
Funcionalidades	Colaboraciones
Insertar profesor	Miembro_polo
Eliminar profesor	
Editar profesor	
Devolver profesor	

Pre-grado	
Funcionalidades	Colaboraciones
Insertar curso_pre-grado	Curso estudiante
Eliminar curso_pre-grado	
Editar curso_pre-grado	
Devolver curso_pre-grado	
Devolver estudiantes dado un curso_pre-grado	
Devolver los curso_pre-grado de un estudiantes	
Devolver los estudiantes que no estan en un curso_pre-grado	

Port-grado	
Funcionalidades	Colaboraciones
Insertar curso_post-grado	Curso Profesor
Eliminar curso_post-grado	
Editar curso_post-grado	
Devolver curso_post-grado	
Devolver profesores dado un curso_post-grado	
Devolver los curso_post-grado de un profesor	
Devolver las superaciones de un profesor	
Devolver los profesores de una superación dada	

Devolver los profesores que no están en un curso_post-grado

<u>Curso</u>	
Funcionalidades	Colaboraciones
Insertar curso	
Eliminar curso	
Editar curso	
Devolver curso	

<u>Superación</u>	
Funcionalidades	Colaboraciones
Insertar superación	
Eliminar superación	
Editar superación	
Devolver superación	

<u>Estudiante pre-grado</u>	
Funcionalidades	Colaboraciones
Asignar un estudiante a un curso pre-grado	Estudiante Pre_grado

<u>Profesor post-grado</u>	
Funcionalidades	Colaboraciones
Asignar un profesor a un curso post-grado	Profesor Post-grado

<u>Profesor superación</u>	
Funcionalidades	Colaboraciones
Asignar un profesor a una superación	Profesor Superación

<u>Superación post-grado</u>	
Funcionalidades	Colaboraciones
Asignar un curso post-grado a una superación	Superación Post-grado

<u>Evento</u>	
Funcionalidades	Colaboraciones
Insertar evento	Miembro_polo
Eliminar evento	
Editar evento	
Devolver evento	
Devolver los eventos de un miembro	
Devolver los miembro de un evento	

<u>Tesis</u>	
Funcionalidades	Colaboraciones
Insertar tesis	Actividad de Investigación
Eliminar tesis	
Editar tesis	
Devolver tesis	

<u>Investigación</u>	
Funcionalidades	Colaboraciones
Insertar investigación	Actividad de Investigación
Eliminar investigación	
Editar investigación	
Devolver investigación	
Devolver las investigaciones de un miembro	
Devolver los miembro de una investigación	

<u>Actividad de Investigación</u>	
Funcionalidades	Colaboraciones
Insertar investigación	
Eliminar investigación	
Editar investigación	

<u>Miembro polo evento</u>	
Funcionalidades	Colaboraciones
Asignar un miembro a un evento	Miembro_polo Evento

<u>Miembro polo actividad investigación</u>	
Funcionalidades	Colaboraciones
Asignar un miembro a una tesis	Miembro_polo Tesis investigación
Asignar un miembro a una investigación	

<u>Proyecto</u>	
Funcionalidades	Colaboraciones
Insertar proyecto	Miembro_polo
Eliminar proyecto	
Editar proyecto	
Devolver proyecto	
Devolver los miembros de un proyecto, especificando los roles que tienen asignados.	
Devolver los datos de un miembro que desempeña.	
Devolver el laboratorio en el que trabaja un proyecto determinado.	
Devolver la cantidad de miembros que tiene un proyecto determinado,	

especificando cuántos son profesores y cuántos son estudiantes y de estos últimos cuántos son por años.	
Devolver la fecha en que fue registrado en el sistema un proyecto determinado.	
Devolver los miembros que no están en proyecto.	
Asignar miembros a un proyecto	

<u>Persona</u>	
Funcionalidades	Colaboraciones
Insertar persona	
Eliminar persona	
Editar persona	
Devolver persona	

<u>Institución</u>	
Funcionalidades	Colaboraciones
Insertar institución	
Eliminar institución	
Editar institución	
Devolver institución	

4.3 DISEÑO DE LA BASE DE DATOS

Diseñar la Base de Datos es algo que no se puede pasar por alto, producto de que uno de sus objetivos fundamentales es brindar la seguridad en el almacenamiento de la información a gestionar.

El modelo de datos de la aplicación en cuestión posee un nivel de complejidad medio, producto que contiene varias relaciones de herencia y a su vez se generan varias tablas debido a las relaciones de mucho a mucho. A continuación se muestra el modelo de datos que se utilizó:

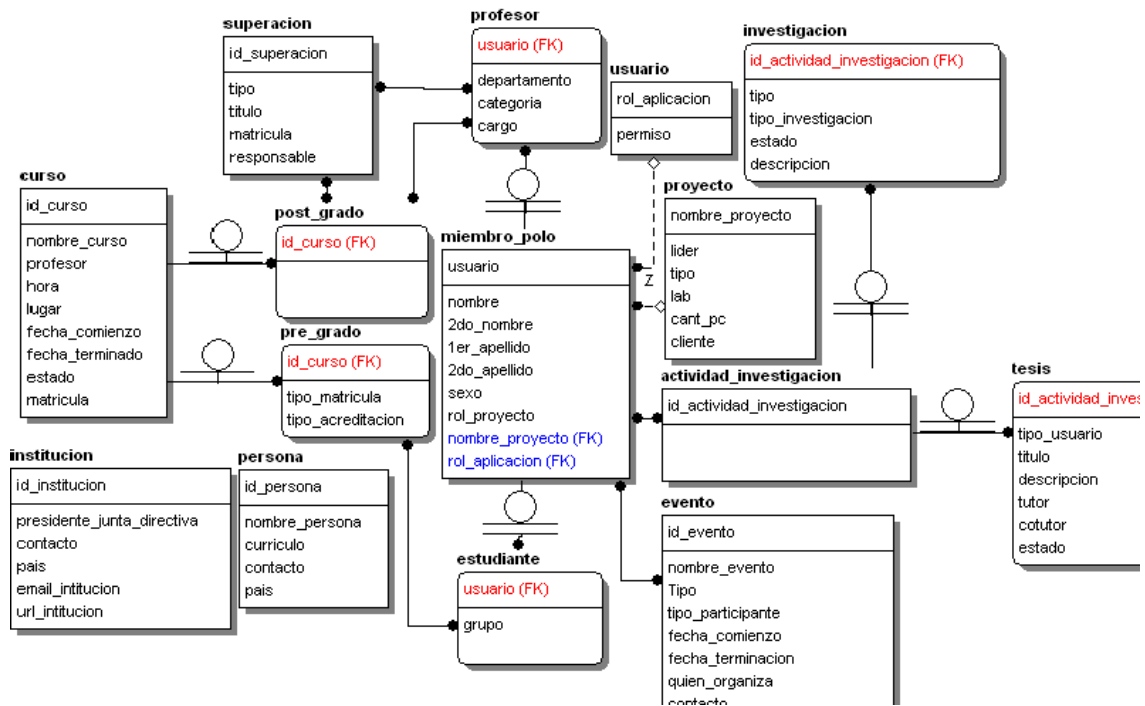


Figura 3 Modelo Lógico

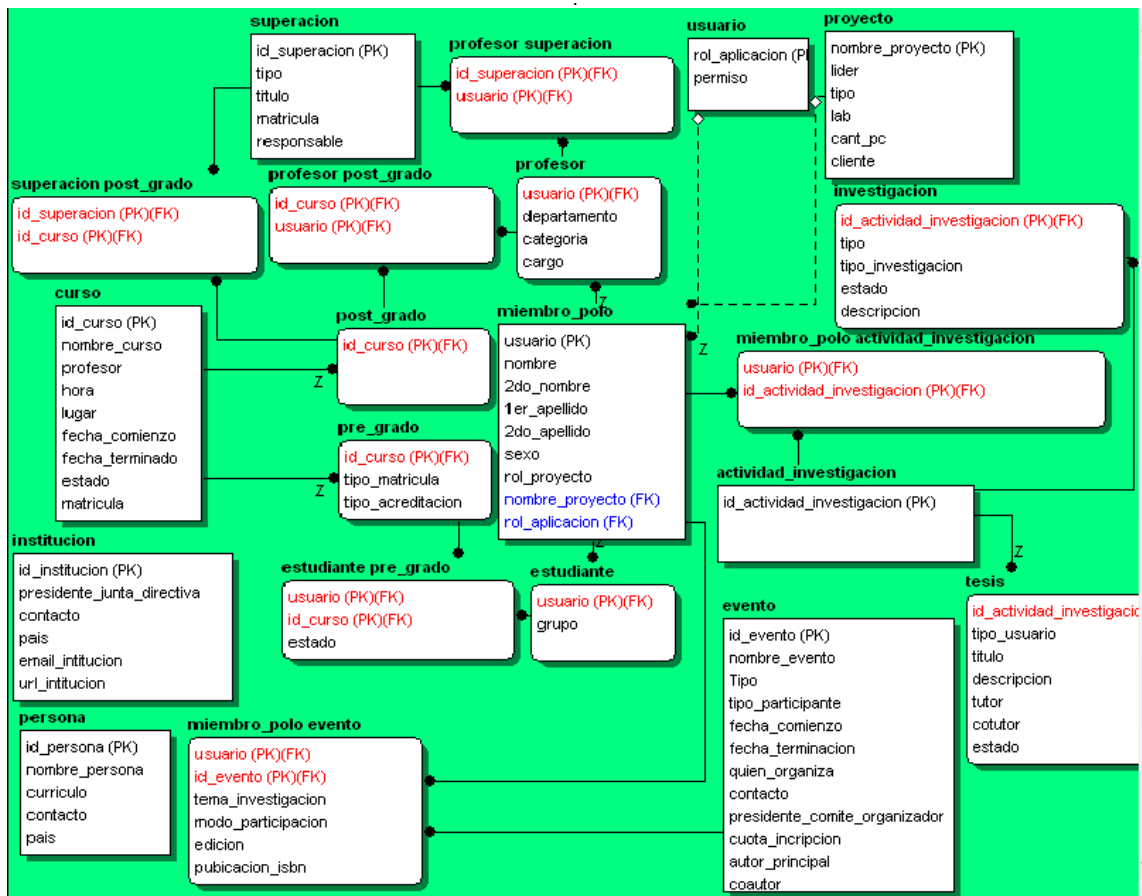


Figura 4 Modelo Físico

4.4 FASE IMPLEMENTACIÓN

Durante el transcurso de las iteraciones se realiza la implementación de las historias de usuario seleccionadas para ser realizadas en cada una de ellas. Al principio de estas se lleva a cabo una revisión del plan de iteraciones y se modifica en caso de ser necesario. Como parte de este plan, se descomponen las historias de usuario en tareas de desarrollo, asignando a un grupo de desarrollo (o una persona), responsable de su implementación. Estas tareas son para el uso estricto de los programadores, pueden ser escritas en lenguaje técnico y no necesariamente entendible por el cliente. Teniendo en cuenta la planificación realizada anteriormente, se llevaron a cabo dos iteraciones de desarrollo sobre el sistema, obteniéndose como finalidad un producto con todas las restricciones y características deseadas para ser utilizado. A continuación se detallan cada una de las iteraciones.

Iteración 1

En esta iteración se implementaron las historias de usuario de mayor prioridad, con el fin de obtener una versión del producto con algunas de las funcionalidades críticas para ser mostrado al cliente y tomar nuevas iniciativas de forma rápida.

Historias de usuario desarrolladas en la iteración 1:

Historia de Usuario	Estimación	Real
1-Gestionar miembro	1	1
2- Autenticar	0.5	0.5
3- Publicar	0.5	0.5

4.4.1 Tareas de las historias de usuario desarrolladas en la iteración 1

Gestionar miembro:

Tarea #1 de la historia de usuario Gestionar miembro.

Tarea	
Número tarea: 1	Número historia: 1
Nombre tarea: Configuración de las interfaces para gestionar miembro.	
Tipo de tarea : Configuración	Puntos estimados: 0.25
Fecha inicio: 13 abril 2009	Fecha fin: 15 abril 2009
Programador responsable: Yan Monteagudo Jiménez y Ernesto Muñoz Castillo	

Descripción:

Se configuran las interfaces para insertar, eliminar, editar y mostrar los datos necesarios de cada usuario que hará uso de la aplicación.

Tarea #2 de la historia de usuario Gestionar miembro.

Tarea	
Número tarea: 2	Número historia: 1
Nombre tarea: Captura de datos del usuario	
Tipo de tarea : Desarrollo	Puntos estimados: 0.25
Fecha inicio: 13 abril 2009	Fecha fin: 16 abril 2009
Programador responsable: Yan Monteagudo Jiménez y Ernesto Muñoz Castillo	
Descripción:	
Se capturan los datos entrados en caso de insertar un nuevo usuario para una posterior comparación o se muestra el listado de usuarios para capturar los datos de un usuario en específico y realizar la acción deseada posteriormente.	

Tarea #3 de la historia de usuario Gestionar miembro.

Tarea	
Número tarea: 3	Número historia: 1
Nombre tarea: Verificación de existencia o no del usuario	
Tipo de tarea : Desarrollo	Puntos estimados: 0.25
Fecha inicio: 15 abril 2009	Fecha fin: 16 abril 2009
Programador responsable: Yan Monteagudo Jiménez y Ernesto Muñoz Castillo	
Descripción:	
Se verificará si el usuario a gestionar existe o no en la base de datos.	

Tarea #4 de la historia de usuario Gestionar miembro.

Tarea	
Número tarea: 4	Número historia: 1
Nombre tarea: Gestión de datos del usuario en la base de datos	
Tipo de tarea : Desarrollo	Puntos estimados: 0.25
Fecha inicio: 16 abril 2009	Fecha fin: 19 abril 2009

Programador responsable: Yan Monteagudo Jiménez y Ernesto Muñoz Castillo

Descripción:

A la hora de ingresar un nuevo usuario si este no existe sus datos se almacenan en la base de datos para lograr gestionar sus datos, en caso de que este usuario exista ya se pueden realizar todas las acciones de gestionar según la opción escogida.

Autenticar:

Tarea #1 de la historia de usuario Autenticar.

Tarea	
Número tarea: 1	Número historia: 2
Nombre tarea: Configuración de la interfaz para la autenticación.	
Tipo de tarea : Configuración	Puntos estimados: 0.12
Fecha inicio: 19 abril 2009	Fecha fin: 20 abril 2009
Programador responsable: Yan Monteagudo Jiménez y Ernesto Muñoz Castillo	
Descripción:	
Se configura la interfaz de inicio en la cual se podrán autenticar los usuarios de la aplicación.	

Tarea #2 de la historia de usuario Autenticar.

Tarea	
Número tarea: 2	Número historia: 2
Nombre tarea: Captura de los datos de autenticación	
Tipo de tarea : Desarrollo	Puntos estimados: 0.12
Fecha inicio: 19 abril 2009	Fecha fin: 21 abril 2009
Programador responsable: Yan Monteagudo Jiménez y Ernesto Muñoz Castillo	
Descripción:	
Se capturan los datos de autenticación de los usuarios para su posterior comprobación.	

Tarea #3 de la historia de usuario Autenticar.

Tarea	
Número tarea: 3	Número historia: 2
Nombre tarea: Comprobar si el usuario es válido	

Tipo de tarea : Desarrollo	Puntos estimados: 0.12
Fecha inicio: 19 abril 2009	Fecha fin: 21 abril 2009
Programador responsable: Yan Monteagudo Jiménez y Ernesto Muñoz Castillo	
Descripción: Se comprueban los datos del usuario autenticado verificando si es válido o no.	

Tarea #4 de la historia de usuario Autenticar.

Tarea	
Número tarea: 4	Número historia: 2
Nombre tarea: Acceso a la aplicación con los permisos correspondientes	
Tipo de tarea : Desarrollo	Puntos estimados: 0.12
Fecha inicio: 20 abril 2009	Fecha fin: 22 abril 2009
Programador responsable: Yan Monteagudo Jiménez y Ernesto Muñoz Castillo	
Descripción: Se le da acceso a la aplicación si el usuario es válido con los permisos asignados según el rol que juega en el sistema.	

Publicar:

Tarea #1 de la historia de usuario Publicar.

Tarea	
Número tarea: 1	Número historia: 3
Nombre tarea: Configuración de la interfaz para la publicación.	
Tipo de tarea : Configuración	Puntos estimados: 0.14
Fecha inicio: 23 abril 2009	Fecha fin: 24 abril 2009
Programador responsable: Yan Monteagudo Jiménez y Ernesto Muñoz Castillo	
Descripción: Se configura la interfaz de publicación para las actividades de formación, actividades de investigación y los proyectos.	

Tarea #2 de la historia de usuario Publicar.

Tarea	
Número tarea: 2	Número historia: 3
Nombre tarea: Captura de datos de la publicación	
Tipo de tarea : Desarrollo	Puntos estimados: 0.14
Fecha inicio: 23 abril 2009	Fecha fin: 25 abril 2009
Programador responsable: Yan Monteagudo Jiménez y Ernesto Muñoz Castillo	
Descripción: Se capturan los datos de la publicación deseada según el rol que juega en el sistema para su posterior acción.	

Tarea #3 de la historia de usuario Publicar.

Tarea	
Número tarea: 3	Número historia: 3
Nombre tarea: Publicar	
Tipo de tarea : Desarrollo	Puntos estimados: 0.14
Fecha inicio: 24 abril 2009	Fecha fin: 26 abril 2009
Programador responsable: Yan Monteagudo Jiménez y Ernesto Muñoz Castillo	
Descripción: Presentación del listado en la interfaz gráfica de la actividad publicada.	

Iteración 2

Con el desarrollo de esta iteración se obtiene una idea completa y general de cómo quedará la aplicación en cuanto a funcionalidades.

Historias de usuario desarrolladas en la iteración 2:

Historia de Usuario	Estimación	Real
1-Gestionar Actividades Formación	1	1
2- Gestionar Actividades Investigación	1	1
3- Gestionar Proyectos	1	1

4.4.2 Tareas de las historias de usuario desarrolladas en la iteración 2

Gestionar Actividades de Formación:

Tarea #1 de la historia de usuario Gestionar Actividades de Formación.

Tarea	
Número tarea: 1	Número historia: 1
Nombre tarea: Configuración de las interfaces para gestionar los datos de cursos, diplomados y maestrías.	
Tipo de tarea : Configuración	Puntos estimados: 0.16
Fecha inicio: 27 abril 2009	Fecha fin: 28 abril 2009
Programador responsable: Yan Monteagudo Jiménez y Ernesto Muñoz Castillo	
Descripción: Se configuran las interfaces para crear, eliminar, editar, mostrar los datos necesarios de cada curso, diplomado o maestría que se manejarán en la aplicación.	

Tarea #2 de la historia de usuario Gestionar Actividades de Formación.

Tarea	
Número tarea: 2	Número historia: 1
Nombre tarea: Capturar datos de un curso, diplomado o maestría	
Tipo de tarea : Desarrollo	Puntos estimados: 0.16
Fecha inicio: 27 abril 2009	Fecha fin: 30 abril 2009
Programador responsable: Yan Monteagudo Jiménez y Ernesto Muñoz Castillo	
Descripción: En caso de crear curso, diplomado o maestría se capturan los datos para una posterior comparación, para editar, eliminar o mostrar datos se muestra un listado para capturar los datos del curso, diplomado o maestría y posteriormente poder ejecutar la acción.	

Tarea #3 de la historia de usuario Gestionar Actividades de Formación.

Tarea	
Número tarea: 3	Número historia: 1
Nombre tarea: Verificación de existencia o no de un curso, diplomado o maestría.	
Tipo de tarea : Desarrollo	Puntos estimados: 0.16
Fecha inicio: 28 abril 2009	Fecha fin: 29 abril 2009
Programador responsable: Yan Monteagudo Jiménez y Ernesto Muñoz Castillo	

Descripción:

Se verificará si el curso, diplomado o maestría a gestionar, existe o no en la base de datos.

Tarea #4 de la historia de usuario Gestionar Actividades de Formación.

Tarea	
Número tarea: 4	Número historia: 1
Nombre tarea: Gestionar los datos del curso, diplomado o maestría en la base de datos.	
Tipo de tarea : Desarrollo	Puntos estimados: 0.16
Fecha inicio: 30 abril 2009	Fecha fin: 3 mayo 2009
Programador responsable: Yan Monteagudo Jiménez y Ernesto Muñoz Castillo	
Descripción:	
Si el curso, diplomado o maestría no existe, sus datos se almacenan en la base de datos para poder gestionarlo posteriormente, en caso contrario se escoge el curso, diplomado o maestría que se quiera gestionar de la lista mostrada y se ejecuta la acción deseada.	

Gestionar Actividades de Investigación:

Tarea #1 de la historia de usuario Gestionar Actividades de Investigación.

Tarea	
Número tarea: 1	Número historia: 2
Nombre tarea: Configuración de las interfaces para gestionar los datos de eventos, tesis e investigaciones.	
Tipo de tarea : Configuración	Puntos estimados: 0.16
Fecha inicio: 4 mayo 2009	Fecha fin: 5 mayo 2009
Programador responsable: Yan Monteagudo Jiménez y Ernesto Muñoz Castillo	
Descripción:	
Se configuran las interfaces para crear, eliminar, editar, mostrar los datos necesarios de cada evento, tesis o investigación que se manejarán en la aplicación.	

Tarea #2 de la historia de usuario Gestionar Actividades de Investigación.

Tarea	
Número tarea: 2	Número historia: 2
Nombre tarea: Capturar datos de un evento, una tesis o investigación	
Tipo de tarea : Desarrollo	Puntos estimados: 0.16
Fecha inicio: 4 mayo 2009	Fecha fin: 7 mayo 2009
Programador responsable: Yan Monteagudo Jiménez y Ernesto Muñoz Castillo	
Descripción:	
En caso de crear evento, tesis o investigación se capturan los datos para una posterior comparación, para editar, eliminar o mostrar datos se muestra un listado para capturar los datos del evento, tesis o investigación y posteriormente poder ejecutar la acción.	

Tarea #3 de la historia de usuario Gestionar Actividades de Investigación.

Tarea	
Número tarea: 3	Número historia: 2
Nombre tarea: Verificación de existencia o no de un evento, una tesis o investigación.	
Tipo de tarea : Desarrollo	Puntos estimados: 0.16
Fecha inicio: 5 mayo 2009	Fecha fin: 6 mayo 2009
Programador responsable: Yan Monteagudo Jiménez y Ernesto Muñoz Castillo	
Descripción:	
Se verificará si el evento, tesis o investigación a gestionar existe o no en la base de datos.	

Tarea #4 de la historia de usuario Gestionar Actividades de Investigación.

Tarea	
Número tarea: 4	Número historia: 2
Nombre tarea: Gestionar los datos del evento, la tesis o investigación en la base de datos.	
Tipo de tarea : Desarrollo	Puntos estimados: 0.16
Fecha inicio: 7 mayo 2009	Fecha fin: 10 mayo 2009
Programador responsable: Yan Monteagudo Jiménez y Ernesto Muñoz Castillo	
Descripción:	
Si el evento, la tesis o investigación no existe, sus datos se almacenan en la base de	

datos para lograr gestionarlo posteriormente, en caso contrario se escoge el evento, la tesis o investigación que se quiera gestionar de la lista mostrada y se ejecuta la acción deseada.

Gestionar Proyecto:

Tarea #1 de la historia de usuario Gestionar Proyecto.

Tarea	
Número tarea: 1	Número historia: 3
Nombre tarea: Configuración de las interfaces para gestionar los datos de los proyectos.	
Tipo de tarea : Configuración	Puntos estimados: 0.16
Fecha inicio: 11 mayo 2009	Fecha fin: 12 mayo 2009
Programador responsable: Yan Monteagudo Jiménez y Ernesto Muñoz Castillo	
Descripción: Se configuran las interfaces para crear, eliminar, editar, mostrar los datos necesarios de cada proyecto que se manejarán en la aplicación.	

Tarea #2 de la historia de usuario Gestionar Proyecto.

Tarea	
Número tarea: 2	Número historia: 3
Nombre tarea: Capturar datos de un proyecto.	
Tipo de tarea : Desarrollo	Puntos estimados: 0.16
Fecha inicio: 11 mayo 2009	Fecha fin: 14 mayo 2009
Programador responsable: Yan Monteagudo Jiménez y Ernesto Muñoz Castillo	
Descripción: En caso de crear un proyecto se capturan los datos para una posterior comparación, para editar, eliminar o mostrar datos se muestra un listado para capturar los datos del proyecto y posteriormente poder ejecutar la acción.	

Tarea #3 de la historia de usuario Gestionar Proyecto.

Tarea	
Número tarea: 3	Número historia: 3
Nombre tarea: Verificación de existencia o no de un proyecto.	

Tipo de tarea : Desarrollo	Puntos estimados: 0.16
Fecha inicio: 12 mayo 2009	Fecha fin: 13 mayo 2009
Programador responsable: Yan Monteagudo Jiménez y Ernesto Muñoz Castillo	
Descripción: Se verificará si el proyecto a gestionar existe o no en la base de datos.	

Tarea #4 de la historia de usuario Gestionar Proyecto.

Tarea	
Número tarea: 4	Número historia: 3
Nombre tarea: Gestionar los datos de un proyecto en la base de datos.	
Tipo de tarea : Desarrollo	Puntos estimados: 0.16
Fecha inicio: 14 mayo 2009	Fecha fin: 17 mayo 2009
Programador responsable: Yan Monteagudo Jiménez y Ernesto Muñoz Castillo	
Descripción: Si el proyecto no existe, sus datos se almacenan en la base de datos para poder gestionarlo posteriormente, en caso contrario se escoge el proyecto que se quiera gestionar de la lista mostrada y se ejecuta la acción deseada.	

4.5 FASE PRUEBA

Uno de los pilares fundamentales de XP es el proceso de pruebas, el cual anima a los desarrolladores a probar constantemente tanto como sea posible. Mediante esta filosofía se reduce el número de errores no detectados así como el tiempo entre la introducción de éste en el sistema y su detección. Todo ésto contribuye a elevar la calidad de los productos desarrollados y a la seguridad de los programadores a la hora de introducir cambios o modificaciones. La metodología XP divide las pruebas en dos grupos: pruebas unitarias, desarrolladas por los programadores, encargadas de verificar el código de forma automática y las pruebas de aceptación, destinadas a evaluar si al final de una iteración se obtuvo la funcionalidad requerida, además de comprobar que dicha funcionalidad sea la esperada por el cliente.

4.5.1 Pruebas de aceptación

Las pruebas de aceptación son pruebas de caja negra que se crean a partir de las historias de usuario. Durante las iteraciones las historias de usuarios seleccionadas

serán traducidas a pruebas de aceptación. En ellas se especifican, desde la perspectiva del cliente, los escenarios para probar que una historia de usuario ha sido implementada correctamente. Una historia de usuario puede tener todas las pruebas de aceptación que necesite para asegurar su correcto funcionamiento. El objetivo final de éstas es garantizar que los requerimientos han sido cumplidos y que el sistema es aceptable. Una historia de usuario no se considera completa hasta que no ha pasado por sus pruebas de aceptación.

4.5.2 Pruebas de Aceptación de la Iteración 1

Historias de Usuarios de la iteración 1:

Historia de Usuario	Estimación	Real
1-Gestionar miembro	1	1
2-Autenticar	0.5	0.5
3- Publicar	0.5	0.5

Gestionar miembro:

Prueba de aceptación para la historia de usuario Gestionar miembro.

Caso de Prueba de Aceptación	
Código: HU1_P1	Historia de Usuario: Gestionar miembro
Nombre: Insertar un estudiante al sistema	
Descripción: Probar que se inserten los datos de un estudiante correctamente en la base de datos.	
Condiciones de Ejecución: La aplicación debe ser ejecutada correctamente por el Jefe del Polo. Se insertaran datos de un estudiante válidos. Deben ser llenados todos los campos para la inserción.	
Entrada / Pasos de ejecución: Se intenta insertar un estudiante con datos válidos.	
Resultado Esperado: El estudiante es registrado correctamente.	
Evaluación de la Prueba: Prueba satisfactoria.	

Prueba de aceptación para la historia de usuario Gestionar miembro.

Caso de Prueba de Aceptación	
Código: HU1_P2	Historia de Usuario: Gestionar miembro

Nombre: Eliminar un estudiante del sistema
Descripción: Probar que se eliminen los datos de un estudiante correctamente de base de datos.
Condiciones de Ejecución: La aplicación debe ser ejecutada correctamente por el Jefe del Polo. Existencia del estudiante a eliminar.
Entrada / Pasos de ejecución: Se escoge un estudiante para eliminar.
Resultado Esperado: El estudiante es eliminado correctamente.
Evaluación de la Prueba: Prueba satisfactoria.

Prueba de aceptación para la historia de usuario Gestionar miembro.

Caso de Prueba de Aceptación	
Código: HU1_P3	Historia de Usuario: Gestionar miembro
Nombre: Editar un estudiante del sistema	
Descripción: Probar que se modifique un estudiante correctamente en la base de datos.	
Condiciones de Ejecución: La aplicación debe ser ejecutada correctamente por el Jefe del Polo. Existencia del estudiante a modificar. Deben ser llenados todos los campos para la modificación.	
Entrada / Pasos de ejecución: Se escoge un estudiante para actualizar sus datos.	
Resultado Esperado: El estudiante es modificado correctamente.	
Evaluación de la Prueba: Prueba satisfactoria.	

Prueba de aceptación para la historia de usuario Gestionar miembro.

Caso de Prueba de Aceptación	
Código: HU1_P4	Historia de Usuario: Gestionar miembro
Nombre: Devolver estudiante en la interfaz	
Descripción: Probar que se visualicen los estudiante correctamente en la interfaz.	
Condiciones de Ejecución: La aplicación debe ser ejecutada correctamente por cualquier miembro del Polo.	

Existencia de los estudiantes a visualizar.

Entrada / Pasos de ejecución: Se escoge los estudiantes a visualizar.

Resultado Esperado: Los usuarios son mostrados correctamente.

Evaluación de la Prueba: Prueba satisfactoria.

Prueba de aceptación para la historia de usuario Gestionar miembro.

Caso de Prueba de Aceptación

Código: HU1_P5

Historia de Usuario: Gestionar miembro

Nombre: Insertar un profesor al sistema

Descripción: Probar que se inserten los datos de un profesor correctamente en la base de datos.

Condiciones de Ejecución:

La aplicación debe ser ejecutada correctamente por el Jefe del Polo.

Se insertaran datos de un profesor válidos.

Deben ser llenados todos los campos para la inserción.

Entrada / Pasos de ejecución: Se intenta insertar un profesor con datos válidos

Resultado Esperado: El profesor es registrado correctamente.

Evaluación de la Prueba: Prueba satisfactoria.

Prueba de aceptación para la historia de usuario Gestionar miembro.

Caso de Prueba de Aceptación

Código: HU1_P6

Historia de Usuario: Gestionar miembro

Nombre: Eliminar un profesor del sistema

Descripción: Probar que se eliminen los datos de un profesor correctamente de base de datos.

Condiciones de Ejecución:

La aplicación debe ser ejecutada correctamente por el Jefe del Polo.

Existencia del profesor a eliminar.

Entrada / Pasos de ejecución: Se escoge un profesor para eliminar.

Resultado Esperado: El profesor es eliminado correctamente.

Evaluación de la Prueba: Prueba satisfactoria.

Prueba de aceptación para la historia de usuario Gestionar miembro.

Caso de Prueba de Aceptación

Código: HU1_P7

Historia de Usuario: Gestionar miembro

Nombre: Editar un profesor del sistema

Descripción: Probar que se modifique un profesor correctamente en la base de datos.

Condiciones de Ejecución:

La aplicación debe ser ejecutada correctamente por el Jefe del Polo.

Existencia del profesor a modificar.

Deben ser llenados todos los campos para la modificación.

Entrada / Pasos de ejecución: Se escoge un profesor para actualizar sus datos.

Resultado Esperado: El profesor es modificado correctamente.

Evaluación de la Prueba: Prueba satisfactoria.

Prueba de aceptación para la historia de usuario Gestionar miembro.

Caso de Prueba de Aceptación

Código: HU1_P8

Historia de Usuario: Gestionar miembro

Nombre: Devolver profesor en la interfaz

Descripción: Probar que se visualicen los profesor correctamente en la interfaz.

Condiciones de Ejecución:

La aplicación debe ser ejecutada correctamente por cualquier miembro del Polo.

Existencia de los profesores a visualizar.

Entrada / Pasos de ejecución: Se escoge los profesores a visualizar.

Resultado Esperado: Los profesores son mostrados correctamente.

Evaluación de la Prueba: Prueba satisfactoria.

Prueba de aceptación para la historia de usuario Gestionar miembro.

Caso de Prueba de Aceptación	
Código: HU1_P9	Historia de Usuario: Gestionar miembro
Nombre: Devolver todos los cursos en que participó un estudiante en un tiempo determinado en la interfaz.	
Descripción: Probar que se muestren todos los cursos de un estudiante en un tiempo determinado correctamente en la interfaz.	
Condiciones de Ejecución: La aplicación debe ser ejecutada correctamente por el Jefe del Polo. Comprobar la existencia de un estudiante en la base de datos.	
Entrada / Pasos de ejecución: Se escoge el estudiante para visualizar todos los cursos en los cuales ha participado en un tiempo determinado.	
Resultado Esperado: Los cursos de un estudiante sean mostrados correctamente.	
Evaluación de la Prueba: Prueba satisfactoria.	

Prueba de aceptación para la historia de usuario Gestionar miembro.

Caso de Prueba de Aceptación	
Código: HU1_P10	Historia de Usuario: Gestionar miembro
Nombre: Devolver todas las actividades de formación en que ha participa un profesor.	
Descripción: Probar que se muestren todas las actividades de formación en que ha participa un profesos correctamente en la interfaz.	
Condiciones de Ejecución: La aplicación debe ser ejecutada correctamente por el Jefe del Polo. Comprobar la existencia de un profesor en la base de datos.	
Entrada / Pasos de ejecución: Se escoge el profesor para visualizar todas las actividades de formación en que ha participa.	
Resultado Esperado: Las actividades de formación en que ha participado un profesor sean mostradas correctamente.	
Evaluación de la Prueba: Prueba satisfactoria.	

Prueba de aceptación para la historia de usuario Gestionar miembro.

Caso de Prueba de Aceptación	
Código: HU1_P11	Historia de Usuario: Gestionar miembro
Nombre: Devolver las actividades de formación en que está matriculado un miembro.	
Descripción: Probar que se muestren todas las actividades de formación en que está matriculado un miembro correctamente en la interfaz.	
Condiciones de Ejecución: La aplicación debe ser ejecutada correctamente por el Jefe del Polo. Comprobar la existencia de un miembro en la base de datos.	
Entrada / Pasos de ejecución: Se escoge el miembro para visualizar todas las actividades de formación en que está matriculado.	
Resultado Esperado: Las actividades de formación en que está matriculado un miembro sean mostradas correctamente.	
Evaluación de la Prueba: Prueba satisfactoria.	

Prueba de aceptación para la historia de usuario Gestionar miembro.

Caso de Prueba de Aceptación	
Código: HU1_P12	Historia de Usuario: Gestionar miembro
Nombre: Devolver los cursos en los que un miembro ha participado como profesor.	
Descripción: Probar que se muestren todos los cursos en los que un miembro ha participado como profesor correctamente en la interfaz.	
Condiciones de Ejecución: La aplicación debe ser ejecutada correctamente por el Jefe del Polo. Comprobar la existencia de un miembro en la base de datos.	
Entrada / Pasos de ejecución: Se escoge el miembro para visualizar todos los cursos en los que ha participado como profesor.	
Resultado Esperado: Los cursos en los que un miembro ha participado como profesor sean mostrados correctamente.	
Evaluación de la Prueba: Prueba satisfactoria.	

Prueba de aceptación para la historia de usuario Gestionar miembro.

Caso de Prueba de Aceptación	
Código: HU1_P13	Historia de Usuario: Gestionar miembro
Nombre: Devolver los eventos en que ha participado un miembro en un período determinado.	
Descripción: Probar que se muestren los eventos en que ha participado un miembro en un período determinado correctamente en la interfaz.	
Condiciones de Ejecución: La aplicación debe ser ejecutada correctamente por el Jefe del Polo. Comprobar la existencia de un miembro en la base de datos.	
Entrada / Pasos de ejecución: Se escoge el miembro para visualizar todos los eventos en que ha participado en un período determinado.	
Resultado Esperado: Los eventos en que ha participado un miembro en un período determinado sean mostrados correctamente.	
Evaluación de la Prueba: Prueba satisfactoria.	

Prueba de aceptación para la historia de usuario Gestionar miembro.

Caso de Prueba de Aceptación	
Código: HU1_P15	Historia de Usuario: Gestionar miembro
Nombre: Devolver las publicaciones que ha hecho un miembro en un período determinado.	
Descripción: Probar que se muestren las publicaciones que ha hecho un miembro en un período determinado correctamente en la interfaz.	
Condiciones de Ejecución: La aplicación debe ser ejecutada correctamente por el Jefe del Polo. Comprobar la existencia de un miembro en la base de datos.	
Entrada / Pasos de ejecución: Se escoge el miembro para visualizar todas las publicaciones que ha hecho en un período determinado.	
Resultado Esperado: Las publicaciones que ha hecho un miembro en un período determinado sean mostradas correctamente.	
Evaluación de la Prueba: Prueba satisfactoria.	

Autenticar Usuario:

Prueba de aceptación para la historia de usuario Autenticar miembro.

Caso de Prueba de Aceptación	
Código: HU2_P1	Historia de Usuario: Autenticar miembro
Nombre: Autenticación de un usuario en el sistema.	
Descripción: Probar que se autentique un usuario satisfactoriamente en el sistema.	
Condiciones de Ejecución: La aplicación debe ser ejecutada correctamente. Existencia del usuario que se va a autenticar en la base de dato.	
Entrada / Pasos de ejecución: Se introducen los datos de autenticación del usuario.	
Resultado Esperado: El usuario acceda satisfactoriamente a la aplicación.	
Evaluación de la Prueba: Prueba satisfactoria.	

Publicar:

Prueba de aceptación para la historia de usuario Publicar.

Caso de Prueba de Aceptación	
Código: HU3_P1	Historia de Usuario: Publicar
Nombre: Publicar actividades del Polo.	
Descripción: Probar que se publique la actividad satisfactoriamente.	
Condiciones de Ejecución: La aplicación debe ser ejecutada por el usuario adecuado para realizar esta funcionalidad. Existencia de la actividad a publicar.	
Entrada / Pasos de ejecución: Se escoge la actividad a publicar.	
Resultado Esperado: La actividad se publique correctamente.	
Evaluación de la Prueba: Prueba satisfactoria.	

4.5.3 Pruebas de Aceptación de la Iteración 2

Historias de Usuarios de la iteración 2:

Historia de Usuario	Estimación	Real
1-Gestionar Actividades Formación	1	1

2- Gestionar Actividades Investigación	1	1
3- Gestionar Proyectos	1	1

Gestionar Actividades de Formación:

Prueba de aceptación para la historia de usuario Gestionar Actividades de Formación.

Caso de Prueba de Aceptación	
Código: HU1_P1	Historia de Usuario: Gestionar Actividades de Formación
Nombre: Insertar un curso de pre_grado en el sistema	
Descripción: Probar que se inserte el curso de pre_grado correctamente en la base de datos.	
Condiciones de Ejecución: La aplicación debe ser ejecutada correctamente por el Jefe de área. El curso de pre_grado no debe estar registrado en la base de datos. Deben ser llenados todos los campos para la inserción.	
Entrada / Pasos de ejecución: Se intenta insertar un curso de pre_grado con datos válidos.	
Resultado Esperado: Sea almacenado el curso de pre_grado correctamente.	
Evaluación de la Prueba: Prueba satisfactoria.	

Prueba de aceptación para la historia de usuario Gestionar Actividades de Formación.

Caso de Prueba de Aceptación	
Código: HU1_P2	Historia de Usuario: Gestionar Actividades de Formación
Nombre: Eliminar un curso de pre_grado del sistema	
Descripción: Probar que se eliminen todos los datos del curso de pre_grado de la base de datos.	
Condiciones de Ejecución: La aplicación debe ser ejecutada correctamente por el Jefe de área. Comprobar la existencia del curso de pre_grado en la base de datos.	
Entrada / Pasos de ejecución: Se intenta eliminar un curso de pre_grado.	
Resultado Esperado: El curso de pre_grado sea eliminado correctamente.	

Evaluación de la Prueba: Prueba satisfactoria.

Prueba de aceptación para la historia de usuario Gestionar Actividades de Formación.

Caso de Prueba de Aceptación	
Código: HU1_P3	Historia de Usuario: Gestionar Actividades de Formación
Nombre: Editar un curso de pre_grado del sistema	
Descripción: Probar que se modifiquen todos los datos del curso de pre_grado de la base de datos.	
Condiciones de Ejecución: La aplicación debe ser ejecutada correctamente por el Jefe de área. Comprobar la existencia del curso de pre_grado en la base de datos. Deben ser llenados todos los campos para la modificación	
Entrada / Pasos de ejecución: Se intenta modificar un curso de pre_grado.	
Resultado Esperado: El curso de pre_grado sea modificado correctamente.	
Evaluación de la Prueba: Prueba satisfactoria.	

Prueba de aceptación para la historia de usuario Gestionar Actividades de Formación.

Caso de Prueba de Aceptación	
Código: HU1_P4	Historia de Usuario: Gestionar Actividades de Formación
Nombre: Devolver un curso de pre_grado en la interfaz.	
Descripción: Probar que se devuelvan todos los datos del curso de pre_grado en la interfaz.	
Condiciones de Ejecución: La aplicación puede ser ejecutada por cualquier miembro del Polo. Comprobar la existencia del curso de pre_grado en la base de datos.	
Entrada / Pasos de ejecución: Se intenta mostrar los datos de un curso de pre_grado en la interfaz.	
Resultado Esperado: Mostrar el curso de pre_grado correctamente.	
Evaluación de la Prueba: Prueba satisfactoria.	

Prueba de aceptación para la historia de usuario Gestionar Actividades de Formación.

Caso de Prueba de Aceptación	
Código: HU1_P5	Historia de Usuario: Gestionar Actividades de Formación
Nombre: Insertar un curso de post_grado en el sistema	
Descripción: Probar que se inserte el curso de post_grado correctamente en la base de datos.	
Condiciones de Ejecución: La aplicación debe ser ejecutada correctamente por el Jefe de área. El curso de post_grado no debe estar registrado en la base de datos. Deben ser llenados todos los campos para la inserción.	
Entrada / Pasos de ejecución: Se intenta insertar un curso de post_grado con datos válidos.	
Resultado Esperado: Sea almacenado el curso de post_grado correctamente.	
Evaluación de la Prueba: Prueba satisfactoria.	

Prueba de aceptación para la historia de usuario Gestionar Actividades de Formación.

Caso de Prueba de Aceptación	
Código: HU1_P6	Historia de Usuario: Gestionar Actividades de Formación
Nombre: Eliminar un curso de post_grado del sistema	
Descripción: Probar que se eliminen todos los datos del curso de post_grado de la base de datos.	
Condiciones de Ejecución: La aplicación debe ser ejecutada correctamente por el Jefe de área. Comprobar la existencia del curso de post_grado en la base de datos.	
Entrada / Pasos de ejecución: Se intenta eliminar un curso de post_grado.	
Resultado Esperado: El curso de post_grado sea eliminado correctamente.	
Evaluación de la Prueba: Prueba satisfactoria.	

Prueba de aceptación para la historia de usuario Gestionar Actividades de Formación.

Caso de Prueba de Aceptación	
Código: HU1_P7	Historia de Usuario: Gestionar Actividades de Formación

Nombre: Editar un curso de post_grado del sistema
Descripción: Probar que se modifiquen todos los datos del curso de post_grado de la base de datos.
Condiciones de Ejecución: La aplicación debe ser ejecutada correctamente por el Jefe de área. Comprobar la existencia del curso de post_grado en la base de datos. Deben ser llenados todos los campos para la modificación.
Entrada / Pasos de ejecución: Se intenta modificar un curso de post_grado.
Resultado Esperado: El curso de post_grado sea modificado correctamente.
Evaluación de la Prueba: Prueba satisfactoria.

Prueba de aceptación para la historia de usuario Gestionar Actividades de Formación.

Caso de Prueba de Aceptación	
Código: HU1_P8	Historia de Usuario: Gestionar Actividades de Formación
Nombre: Devolver un curso de post_grado en la interfaz.	
Descripción: Probar que se devuelvan todos los datos del curso de post_grado en la interfaz.	
Condiciones de Ejecución: La aplicación puede ser ejecutada por cualquier miembro del Polo. Comprobar la existencia del curso de post_grado en la base de datos.	
Entrada / Pasos de ejecución: Se intenta mostrar los datos de un curso de post_grado en la interfaz.	
Resultado Esperado: Mostrar el curso de post_grado correctamente.	
Evaluación de la Prueba: Prueba satisfactoria.	

Prueba de aceptación para la historia de usuario Gestionar Actividades de Formación.

Caso de Prueba de Aceptación	
Código: HU1_P9	Historia de Usuario: Gestionar Actividades de Formación
Nombre: Insertar un diplomado en el sistema	
Descripción: Probar que se inserte el diplomado correctamente en la base de datos.	

Condiciones de Ejecución:

La aplicación debe ser ejecutada correctamente por el Jefe de área.

El diplomado no debe estar registrado en la base de datos.

Deben ser llenados todos los campos para la inserción.

Entrada / Pasos de ejecución: Se intenta insertar un diplomado con datos válidos.

Resultado Esperado: Sea almacenado el diplomado correctamente.

Evaluación de la Prueba: Prueba satisfactoria.

Prueba de aceptación para la historia de usuario Gestionar Actividades de Formación.

Caso de Prueba de Aceptación

Código: HU1_P10 **Historia de Usuario:** Gestionar Actividades de Formación

Nombre: Eliminar un diplomado del sistema

Descripción: Probar que se eliminen todos los datos del diplomado de la base de datos.

Condiciones de Ejecución:

La aplicación debe ser ejecutada correctamente por el Jefe de área.

Comprobar la existencia del diplomado en la base de datos.

Entrada / Pasos de ejecución: Se intenta eliminar un diplomado.

Resultado Esperado: El diplomado sea eliminado correctamente.

Evaluación de la Prueba: Prueba satisfactoria.

Prueba de aceptación para la historia de usuario Gestionar Actividades de Formación.

Caso de Prueba de Aceptación

Código: HU1_P11 **Historia de Usuario:** Gestionar Actividades de Formación

Nombre: Editar un diplomado del sistema

Descripción: Probar que se modifiquen todos los datos del diplomado de la base de datos.

Condiciones de Ejecución:

La aplicación debe ser ejecutada correctamente por el Jefe de área.

Comprobar la existencia del diplomado en la base de datos.

Deben ser llenados todos los campos para la modificación.

Entrada / Pasos de ejecución: Se intenta modificar un diplomado.

Resultado Esperado: El diplomado sea modificado correctamente.

Evaluación de la Prueba: Prueba satisfactoria.

Prueba de aceptación para la historia de usuario Gestionar Actividades de Formación.

Caso de Prueba de Aceptación

Código: HU1_P12 **Historia de Usuario:** Gestionar Actividades de Formación

Nombre: Devolver un diplomado en la interfaz.

Descripción: Probar que se devuelvan todos los datos del diplomado en la interfaz.

Condiciones de Ejecución:

La aplicación puede ser ejecutada por cualquier miembro del Polo.

Comprobar la existencia del diplomado en la base de datos.

Entrada / Pasos de ejecución: Se intenta mostrar los datos de un diplomado en la interfaz.

Resultado Esperado: Mostrar el diplomado correctamente.

Evaluación de la Prueba: Prueba satisfactoria.

Prueba de aceptación para la historia de usuario Gestionar Actividades de Formación.

Caso de Prueba de Aceptación

Código: HU1_P13 **Historia de Usuario:** Gestionar Actividades de Formación

Nombre: Insertar una maestría en el sistema

Descripción: Probar que se inserte una maestría correctamente en la base de datos.

Condiciones de Ejecución:

La aplicación debe ser ejecutada correctamente por el Jefe de área.

La maestría no debe estar registrada en la base de datos.

Deben ser llenados todos los campos para la inserción.

Entrada / Pasos de ejecución: Se intenta insertar una maestría con datos válidos.

Resultado Esperado: Sea almacenada la maestría correctamente.

Evaluación de la Prueba: Prueba satisfactoria.

Prueba de aceptación para la historia de usuario Gestionar Actividades de Formación.

Caso de Prueba de Aceptación

Código: HU1_P14 **Historia de Usuario:** Gestionar Actividades de Formación

Nombre: Eliminar una maestría del sistema

Descripción: Probar que se eliminen todos los datos de la maestría de la base de datos.

Condiciones de Ejecución:

La aplicación debe ser ejecutada correctamente por el Jefe de área.

Comprobar la existencia de la maestría en la base de datos.

Entrada / Pasos de ejecución: Se intenta eliminar una maestría.

Resultado Esperado: La maestría sea eliminada correctamente.

Evaluación de la Prueba: Prueba satisfactoria.

Prueba de aceptación para la historia de usuario Gestionar Actividades de Formación.

Caso de Prueba de Aceptación

Código: HU1_P15 **Historia de Usuario:** Gestionar Actividades de Formación

Nombre: Editar una maestría del sistema

Descripción: Probar que se modifiquen todos los datos de la maestría de la base de datos.

Condiciones de Ejecución:

La aplicación debe ser ejecutada correctamente por el Jefe de área.

Comprobar la existencia de la maestría en la base de datos.

Deben ser llenados todos los campos para la modificación.

Entrada / Pasos de ejecución: Se intenta modificar una maestría.

Resultado Esperado: La maestría sea modificada correctamente.

Evaluación de la Prueba: Prueba satisfactoria.

Prueba de aceptación para la historia de usuario Gestionar Actividades de Formación.

Caso de Prueba de Aceptación	
Código: HU1_P16	Historia de Usuario: Gestionar Actividades de Formación
Nombre: Devolver una maestría en la interfaz.	
Descripción: Probar que se devuelvan todos los datos de la maestría en la interfaz.	
Condiciones de Ejecución: La aplicación puede ser ejecutada por cualquier miembro del Polo. Comprobar la existencia de la maestría en la base de datos.	
Entrada / Pasos de ejecución: Se intenta mostrar los datos de una maestría en la interfaz.	
Resultado Esperado: Mostrar la maestría correctamente.	
Evaluación de la Prueba: Prueba satisfactoria.	

Prueba de aceptación para la historia de usuario Gestionar Actividades de Formación.

Caso de Prueba de Aceptación	
Código: HU1_P17	Historia de Usuario: Gestionar Actividades de Formación
Nombre: Devolver los estudiantes de un curso de pre-grado en la interfaz.	
Descripción: Probar que se devuelvan todos los estudiantes del curso de pre-grado en la interfaz.	
Condiciones de Ejecución: La aplicación puede ser ejecutada por cualquier miembro del Polo. Comprobar la existencia del curso de pre-grado en la base de datos.	
Entrada / Pasos de ejecución: Se intenta mostrar los estudiantes de un curso de pre-grado en la interfaz.	
Resultado Esperado: Mostrar los estudiantes de un curso de pre-grado correctamente.	
Evaluación de la Prueba: Prueba satisfactoria.	

Prueba de aceptación para la historia de usuario Gestionar Actividades de Formación.

Caso de Prueba de Aceptación	
Código: HU1_P18	Historia de Usuario: Gestionar Actividades de Formación
Nombre: Devolver los cursos de pre-grado de un estudiante en la interfaz.	
Descripción: Probar que se devuelvan todos los cursos de pre-grado de un estudiante en la interfaz.	
Condiciones de Ejecución: La aplicación puede ser ejecutada por cualquier miembro del Polo. Comprobar la existencia del estudiante en la base de datos.	
Entrada / Pasos de ejecución: Se intenta mostrar los cursos de pre-grado de un estudiante en la interfaz.	
Resultado Esperado: Mostrar los cursos de pre-grado de un estudiante correctamente.	
Evaluación de la Prueba: Prueba satisfactoria.	

Prueba de aceptación para la historia de usuario Gestionar Actividades de Formación.

Caso de Prueba de Aceptación	
Código: HU1_P19	Historia de Usuario: Gestionar Actividades de Formación
Nombre: Devolver los estudiante que no están en un cursos de pre-grado en la interfaz.	
Descripción: Probar que se devuelvan todos los estudiantes que no están en un cursos de pre-grado en la interfaz.	
Condiciones de Ejecución: La aplicación puede ser ejecutada por cualquier miembro del Polo. Comprobar la existencia del curso de pre-grado en la base de datos.	
Entrada / Pasos de ejecución: Se intenta mostrar los estudiantes que no están en un curso de pre-grado en la interfaz.	
Resultado Esperado: Mostrar los estudiantes que no están en un cursos de pre-grado correctamente.	
Evaluación de la Prueba: Prueba satisfactoria.	

Prueba de aceptación para la historia de usuario Gestionar Actividades de Formación.

Caso de Prueba de Aceptación	
Código: HU1_P20	Historia de Usuario: Gestionar Actividades de Formación
Nombre: Devolver los profesores de un curso de post-grado en la interfaz.	
Descripción: Probar que se devuelvan todos los profesores del curso de post-grado en la interfaz.	
Condiciones de Ejecución: La aplicación puede ser ejecutada por cualquier miembro del Polo. Comprobar la existencia del curso de post-grado en la base de datos.	
Entrada / Pasos de ejecución: Se intenta mostrar los profesores de un curso de post-grado en la interfaz.	
Resultado Esperado: Mostrar los profesores de un curso de post-grado correctamente.	
Evaluación de la Prueba: Prueba satisfactoria.	

Prueba de aceptación para la historia de usuario Gestionar Actividades de Formación.

Caso de Prueba de Aceptación	
Código: HU1_P21	Historia de Usuario: Gestionar Actividades de Formación
Nombre: Devolver los cursos de post-grado de un profesor en la interfaz.	
Descripción: Probar que se devuelvan todos los cursos de post-grado de un profesor en la interfaz.	
Condiciones de Ejecución: La aplicación puede ser ejecutada por cualquier miembro del Polo. Comprobar la existencia del profesor en la base de datos.	
Entrada / Pasos de ejecución: Se intenta mostrar los cursos de post-grado de un profesor en la interfaz.	
Resultado Esperado: Mostrar los cursos de post-grado de un profesor correctamente.	
Evaluación de la Prueba: Prueba satisfactoria.	

Prueba de aceptación para la historia de usuario Gestionar Actividades de Formación.

Caso de Prueba de Aceptación	
Código: HU1_P22	Historia de Usuario: Gestionar Actividades de Formación
Nombre: Devolver los profesores que no están en un cursos de post-grado en la interfaz.	
Descripción: Probar que se devuelvan todos los profesores que no están en un cursos de post-grado en la interfaz.	
Condiciones de Ejecución: La aplicación puede ser ejecutada por cualquier miembro del Polo. Comprobar la existencia del curso de post-grado en la base de datos.	
Entrada / Pasos de ejecución: Se intenta mostrar los profesores que no están en un curso de post-grado en la interfaz.	
Resultado Esperado: Mostrar los profesores que no están en un cursos de post-grado correctamente.	
Evaluación de la Prueba: Prueba satisfactoria.	

Prueba de aceptación para la historia de usuario Gestionar Actividades de Formación.

Caso de Prueba de Aceptación	
Código: HU1_P23	Historia de Usuario: Gestionar Actividades de Formación
Nombre: Devolver los profesores de una superación en la interfaz.	
Descripción: Probar que se devuelvan todos los profesores de una superación en la interfaz.	
Condiciones de Ejecución: La aplicación puede ser ejecutada por cualquier miembro del Polo. Comprobar la existencia de la superación en la base de datos.	
Entrada / Pasos de ejecución: Se intenta mostrar los profesores de una superación en la interfaz.	
Resultado Esperado: Mostrar los profesores de una superación correctamente.	
Evaluación de la Prueba: Prueba satisfactoria.	

Prueba de aceptación para la historia de usuario Gestionar Actividades de Formación.

Caso de Prueba de Aceptación	
Código: HU1_P24	Historia de Usuario: Gestionar Actividades de Formación
Nombre: Devolver las superaciones de un profesor en la interfaz.	
Descripción: Probar que se devuelvan todas las superaciones de un profesor en la interfaz.	
Condiciones de Ejecución: La aplicación puede ser ejecutada por cualquier miembro del Polo. Comprobar la existencia del profesor en la base de datos.	
Entrada / Pasos de ejecución: Se intenta mostrar las superaciones de un profesor en la interfaz.	
Resultado Esperado: Mostrar las superaciones de un profesor correctamente.	
Evaluación de la Prueba: Prueba satisfactoria.	

Gestionar Actividades de Investigación:

Prueba de aceptación para la historia de usuario Gestionar Actividades de Investigación.

Caso de Prueba de Aceptación	
Código: HU2_P1	Historia de Usuario: Gestionar Actividades de Investigación
Nombre: Insertar un evento en el sistema	
Descripción: Probar que se inserte un evento correctamente en la base de datos.	
Condiciones de Ejecución: La aplicación debe ser ejecutada correctamente por el Jefe de área. El evento no debe estar registrado en la base de datos. Deben ser llenados todos los campos para la inserción.	
Entrada / Pasos de ejecución: Se intenta insertar un evento con datos válidos.	
Resultado Esperado: Sea almacenado el evento correctamente.	
Evaluación de la Prueba: Prueba satisfactoria.	

Prueba de aceptación para la historia de usuario Gestionar Actividades de Investigación.

Caso de Prueba de Aceptación	
Código: HU2_P2	Historia de Usuario: Gestionar Actividades de Investigación
Nombre: Eliminar un evento del sistema	
Descripción: Probar que se eliminen todos los datos de un evento de la base de datos.	
Condiciones de Ejecución: La aplicación debe ser ejecutada correctamente por el Jefe de área. Comprobar la existencia del evento en la base de datos.	
Entrada / Pasos de ejecución: Se intenta eliminar un evento.	
Resultado Esperado: El evento sea eliminado correctamente.	
Evaluación de la Prueba: Prueba satisfactoria.	

Prueba de aceptación para la historia de usuario Gestionar Actividades de Investigación.

Caso de Prueba de Aceptación	
Código: HU2_P3	Historia de Usuario: Gestionar Actividades de Investigación
Nombre: Editar un evento del sistema	
Descripción: Probar que se modifiquen todos los datos del evento de la base de datos.	
Condiciones de Ejecución: La aplicación debe ser ejecutada correctamente por el Jefe de área. Comprobar la existencia del evento en la base de datos. Deben ser llenados todos los campos para la modificación.	
Entrada / Pasos de ejecución: Se intenta modificar un evento.	
Resultado Esperado: El evento sea modificado correctamente.	
Evaluación de la Prueba: Prueba satisfactoria.	

Prueba de aceptación para la historia de usuario Gestionar Actividades de Investigación.

Caso de Prueba de Aceptación	
Código: HU2_P4	Historia de Usuario: Gestionar Actividades de Investigación
Nombre: Devolver un evento en la interfaz.	
Descripción: Probar que se devuelvan todos los datos del evento en la interfaz.	
Condiciones de Ejecución: La aplicación puede ser ejecutada por cualquier miembro del Polo. Comprobar la existencia del evento en la base de datos.	
Entrada / Pasos de ejecución: Se intenta mostrar los datos de un evento en la interfaz.	
Resultado Esperado: Mostrar el evento correctamente.	
Evaluación de la Prueba: Prueba satisfactoria.	

Prueba de aceptación para la historia de usuario Gestionar Actividades de Investigación.

Caso de Prueba de Aceptación	
Código: HU2_P5	Historia de Usuario: Gestionar Actividades de Investigación
Nombre: Insertar una tesis en el sistema	
Descripción: Probar que se inserte una tesis correctamente en la base de datos.	
Condiciones de Ejecución: La aplicación debe ser ejecutada correctamente por el Jefe de área. La tesis no debe estar registrada en la base de datos. Deben ser llenados todos los campos para la inserción.	
Entrada / Pasos de ejecución: Se intenta insertar una tesis con datos válidos.	
Resultado Esperado: Sea almacenada la tesis correctamente.	
Evaluación de la Prueba: Prueba satisfactoria.	

Prueba de aceptación para la historia de usuario Gestionar Actividades de Investigación.

Caso de Prueba de Aceptación	
Código: HU2_P6	Historia de Usuario: Gestionar Actividades de Investigación
Nombre: Eliminar una tesis del sistema	
Descripción: Probar que se eliminen todos los datos de una tesis de la base de datos.	
Condiciones de Ejecución: La aplicación debe ser ejecutada correctamente por el Jefe de área. Comprobar la existencia de la tesis en la base de datos.	
Entrada / Pasos de ejecución: Se intenta eliminar una tesis.	
Resultado Esperado: La tesis sea eliminada correctamente.	
Evaluación de la Prueba: Prueba satisfactoria.	

Prueba de aceptación para la historia de usuario Gestionar Actividades de Investigación.

Caso de Prueba de Aceptación	
Código: HU2_P7	Historia de Usuario: Gestionar Actividades de Investigación
Nombre: Editar una tesis del sistema	
Descripción: Probar que se modifiquen todos los datos de una tesis de la base de datos.	
Condiciones de Ejecución: La aplicación debe ser ejecutada correctamente por el Jefe de área. Comprobar la existencia de la tesis en la base de datos. Deben ser llenados todos los campos para la modificación.	
Entrada / Pasos de ejecución: Se intenta modificar una tesis.	
Resultado Esperado: La tesis sea modificada correctamente.	
Evaluación de la Prueba: Prueba satisfactoria.	

Prueba de aceptación para la historia de usuario Gestionar Actividades de Investigación.

Caso de Prueba de Aceptación	
Código: HU2_P8	Historia de Usuario: Gestionar Actividades de Investigación
Nombre: Devolver una tesis en la interfaz.	
Descripción: Probar que se devuelvan todos los datos de una tesis en la interfaz.	
Condiciones de Ejecución: La aplicación puede ser ejecutada por cualquier miembro del Polo. Comprobar la existencia de la tesis en la base de datos.	
Entrada / Pasos de ejecución: Se intenta mostrar los datos de una tesis en la interfaz.	
Resultado Esperado: Mostrar la tesis correctamente.	
Evaluación de la Prueba: Prueba satisfactoria.	

Prueba de aceptación para la historia de usuario Gestionar Actividades de Investigación.

Caso de Prueba de Aceptación	
Código: HU2_P9	Historia de Usuario: Gestionar Actividades de Investigación
Nombre: Insertar una investigación en el sistema	
Descripción: Probar que se inserte una investigación correctamente en la base de datos.	
Condiciones de Ejecución: La aplicación debe ser ejecutada correctamente por el Jefe de área. La investigación no debe estar registrada en la base de datos. Deben ser llenados todos los campos para la inserción.	
Entrada / Pasos de ejecución: Se intenta insertar una investigación con datos válidos.	
Resultado Esperado: Sea almacenada la investigación correctamente.	
Evaluación de la Prueba: Prueba satisfactoria.	

Prueba de aceptación para la historia de usuario Gestionar Actividades de Investigación.

Caso de Prueba de Aceptación	
Código: HU2_P10	Historia de Usuario: Gestionar Actividades de Investigación
Nombre: Eliminar una investigación del sistema	
Descripción: Probar que se eliminen todos los datos de una investigación de la base de datos.	
Condiciones de Ejecución: La aplicación debe ser ejecutada correctamente por el Jefe de área. Comprobar la existencia de la investigación en la base de datos.	
Entrada / Pasos de ejecución: Se intenta eliminar una investigación.	
Resultado Esperado: La investigación sea eliminada correctamente.	
Evaluación de la Prueba: Prueba satisfactoria.	

Prueba de aceptación para la historia de usuario Gestionar Actividades de Investigación.

Caso de Prueba de Aceptación	
Código: HU2_P11	Historia de Usuario: Gestionar Actividades de Investigación
Nombre: Editar una investigación del sistema	
Descripción: Probar que se modifiquen todos los datos de una investigación de la base de datos.	
Condiciones de Ejecución: La aplicación debe ser ejecutada correctamente por el Jefe de área. Comprobar la existencia de la investigación en la base de datos. Deben ser llenados todos los campos para la modificación.	
Entrada / Pasos de ejecución: Se intenta modificar una investigación.	
Resultado Esperado: La investigación sea modificada correctamente.	
Evaluación de la Prueba: Prueba satisfactoria.	

Prueba de aceptación para la historia de usuario Gestionar Actividades de Investigación.

Caso de Prueba de Aceptación	
Código: HU2_P12	Historia de Usuario: Gestionar Actividades de Investigación
Nombre: Devolver investigación en la interfaz.	
Descripción: Probar que se devuelvan todos los datos de una investigación en la interfaz.	
Condiciones de Ejecución: La aplicación puede ser ejecutada por cualquier miembro del Polo. Comprobar la existencia de la investigación en la base de datos.	
Entrada / Pasos de ejecución: Se intenta mostrar los datos de una investigación en la interfaz.	
Resultado Esperado: Mostrar la investigación correctamente.	
Evaluación de la Prueba: Prueba satisfactoria.	

Prueba de aceptación para la historia de usuario Gestionar Actividades de Investigación.

Caso de Prueba de Aceptación	
Código: HU2_P13	Historia de Usuario: Gestionar Actividades de Investigación
Nombre: Devolver los eventos de un miembro en la interfaz.	
Descripción: Probar que se devuelvan todos los eventos de un miembro en la interfaz.	
Condiciones de Ejecución: La aplicación puede ser ejecutada por cualquier miembro del Polo. Comprobar la existencia del miembro en la base de datos.	
Entrada / Pasos de ejecución: Se intenta mostrar los eventos de un miembro en la interfaz.	
Resultado Esperado: Mostrar los eventos de un miembro correctamente.	
Evaluación de la Prueba: Prueba satisfactoria.	

Prueba de aceptación para la historia de usuario Gestionar Actividades de Investigación.

Caso de Prueba de Aceptación	
Código: HU2_P14	Historia de Usuario: Gestionar Actividades de Investigación
Nombre: Devolver los miembro de un evento en la interfaz.	
Descripción: Probar que se devuelvan todos los miembros de un evento en la interfaz.	
Condiciones de Ejecución: La aplicación puede ser ejecutada por cualquier miembro del Polo. Comprobar la existencia del evento en la base de datos.	
Entrada / Pasos de ejecución: Se intenta mostrar los miembros de un evento en la interfaz.	
Resultado Esperado: Mostrar los miembros de un evento correctamente.	
Evaluación de la Prueba: Prueba satisfactoria.	

Prueba de aceptación para la historia de usuario Gestionar Actividades de Investigación.

Caso de Prueba de Aceptación	
Código: HU2_P15	Historia de Usuario: Gestionar Actividades de Investigación
Nombre: Devolver los miembro de una investigación en la interfaz.	
Descripción: Probar que se devuelvan todos los miembros de una investigación en la interfaz.	
Condiciones de Ejecución: La aplicación puede ser ejecutada por cualquier miembro del Polo. Comprobar la existencia de la investigación en la base de datos.	
Entrada / Pasos de ejecución: Se intenta mostrar los miembros de una investigación en la interfaz.	
Resultado Esperado: Mostrar los miembros de una investigación correctamente.	
Evaluación de la Prueba: Prueba satisfactoria.	

Prueba de aceptación para la historia de usuario Gestionar Actividades de Investigación.

Caso de Prueba de Aceptación	
Código: HU2_P16	Historia de Usuario: Gestionar Actividades de Investigación
Nombre: Devolver las investigaciones de un miembro en la interfaz.	
Descripción: Probar que se devuelvan todas las investigaciones de un miembros en la interfaz.	
Condiciones de Ejecución: La aplicación puede ser ejecutada por cualquier miembro del Polo. Comprobar la existencia del miembro en la base de datos.	
Entrada / Pasos de ejecución: Se intenta mostrar las investigaciones de un miembro en la interfaz.	
Resultado Esperado: Mostrar las investigaciones de un miembro correctamente.	
Evaluación de la Prueba: Prueba satisfactoria.	

Gestionar Proyecto:

Prueba de aceptación para la historia de usuario Gestionar Proyecto.

Caso de Prueba de Aceptación	
Código: HU3_P1	Historia de Usuario: Gestionar Proyecto
Nombre: Insertar un proyecto en el sistema	
Descripción: Probar que se inserte un proyecto correctamente en la base de datos.	
Condiciones de Ejecución: La aplicación debe ser ejecutada correctamente por el Jefe de Polo. El proyecto no debe estar registrado en la base de datos. Deben ser llenados todos los campos para la inserción.	
Entrada / Pasos de ejecución: Se intenta insertar un proyecto con datos válidos.	
Resultado Esperado: Sea almacenado el proyecto correctamente.	
Evaluación de la Prueba: Prueba satisfactoria.	

Prueba de aceptación para la historia de usuario Gestionar Proyecto.

Caso de Prueba de Aceptación	
Código: HU3_P2	Historia de Usuario: Gestionar Proyecto
Nombre: Eliminar un proyecto del sistema	
Descripción: Probar que se eliminen todos los datos de un proyecto de la base de datos.	
Condiciones de Ejecución: La aplicación debe ser ejecutada correctamente por el Jefe de Polo. Comprobar la existencia del proyecto en la base de datos.	
Entrada / Pasos de ejecución: Se intenta eliminar un proyecto.	
Resultado Esperado: El proyecto sea eliminado correctamente.	
Evaluación de la Prueba: Prueba satisfactoria.	

Prueba de aceptación para la historia de usuario Gestionar Proyecto.

Caso de Prueba de Aceptación	
Código: HU3_P3	Historia de Usuario: Gestionar Proyecto
Nombre: Editar un proyecto del sistema	
Descripción: Probar que se modifiquen todos los datos del proyecto de la base de datos.	
Condiciones de Ejecución: La aplicación debe ser ejecutada correctamente por el Jefe de Polo. Comprobar la existencia del proyecto en la base de datos. Deben ser llenados todos los campos para la modificación.	
Entrada / Pasos de ejecución: Se intenta modificar un proyecto.	
Resultado Esperado: El proyecto sea modificado correctamente.	
Evaluación de la Prueba: Prueba satisfactoria.	

Prueba de aceptación para la historia de usuario Gestionar Proyecto.

Caso de Prueba de Aceptación	
Código: HU3_P4	Historia de Usuario: Gestionar Proyecto
Nombre: Devolver un proyecto en la interfaz.	

Descripción: Probar que se devuelvan todos los datos del proyecto en la interfaz.
Condiciones de Ejecución: La aplicación puede ser ejecutada por cualquier miembro del Polo. Comprobar la existencia del proyecto en la base de datos.
Entrada / Pasos de ejecución: Se intenta mostrar los datos de un proyecto en la interfaz.
Resultado Esperado: Mostrar el proyecto correctamente.
Evaluación de la Prueba: Prueba satisfactoria.

Prueba de aceptación para la historia de usuario Gestionar Proyecto.

Caso de Prueba de Aceptación	
Código: HU3_P5	Historia de Usuario: Gestionar Proyecto
Nombre: Devolver los miembros de un proyecto, especificando los roles que tienen asignados y si es estudiante o profesor.	
Descripción: Probar que se devuelvan todos los miembros de un proyecto en la interfaz.	
Condiciones de Ejecución: La aplicación debe ser ejecutada correctamente por el Jefe de Polo. Comprobar la existencia del proyecto en la base de datos. Comprobar la existencia de miembros dentro del proyecto.	
Entrada / Pasos de ejecución: Se intenta mostrar los datos de los miembros de un proyecto en la interfaz.	
Resultado Esperado: Mostrar los miembros de un proyecto correctamente.	
Evaluación de la Prueba: Prueba satisfactoria.	

Prueba de aceptación para la historia de usuario Gestionar Proyecto.

Caso de Prueba de Aceptación	
Código: HU3_P6	Historia de Usuario: Gestionar Proyecto
Nombre: Devolver los datos de un miembro que desempeña.	
Descripción: Probar que se devuelvan los datos de un miembro que desempeña en la	

interfaz.
Condiciones de Ejecución: La aplicación debe ser ejecutada correctamente por el Jefe de Polo. Comprobar la existencia del proyecto en la base de datos. Comprobar la existencia de miembros dentro del proyecto.
Entrada / Pasos de ejecución: Se intenta mostrar los datos de un miembro que desempeña en la interfaz.
Resultado Esperado: Mostrar los datos de un miembro que desempeña correctamente.
Evaluación de la Prueba: Prueba satisfactoria.

Prueba de aceptación para la historia de usuario Gestionar Proyecto.

Caso de Prueba de Aceptación	
Código: HU3_P7	Historia de Usuario: Gestionar Proyecto
Nombre: Devolver el laboratorio en el que trabaja un proyecto determinado.	
Descripción: Probar que se devuelva el laboratorio en el que trabaja un proyecto determinado en la interfaz.	
Condiciones de Ejecución: La aplicación debe ser ejecutada correctamente por el Jefe de Polo. Comprobar la existencia del proyecto en la base de datos. Comprobar si tiene laboratorio asignado.	
Entrada / Pasos de ejecución: Se intenta mostrar el laboratorio en el que trabaja un proyecto determinado en la interfaz.	
Resultado Esperado: Mostrar el laboratorio en el que trabaja un proyecto determinado correctamente.	
Evaluación de la Prueba: Prueba satisfactoria.	

Prueba de aceptación para la historia de usuario Gestionar Proyecto.

Caso de Prueba de Aceptación	
Código: HU3_P8	Historia de Usuario: Gestionar Proyecto
Nombre: Devolver la cantidad de miembros que tiene un proyecto determinado	

especificando cuántos son profesores y cuántos son estudiantes y de estos últimos cuántos son por años.

Descripción: Probar que se devuelva la cantidad de miembros que tiene un proyecto determinado especificando cuántos son profesores y cuántos son estudiantes y de estos últimos cuántos son por años en la interfaz.

Condiciones de Ejecución:

La aplicación debe ser ejecutada correctamente por el Jefe de Polo.

Comprobar la existencia del proyecto en la base de datos.

Comprobar la existencia de miembros dentro del proyecto.

Entrada / Pasos de ejecución: Se intenta mostrar la cantidad de miembros que tiene un proyecto determinado especificando cuántos son profesores y cuántos son estudiantes y de estos últimos cuántos son por años en la interfaz.

Resultado Esperado: Mostrar la cantidad de miembros que tiene un proyecto determinado especificando cuántos son profesores y cuántos son estudiantes y de estos últimos cuántos son por años correctamente.

Evaluación de la Prueba: Prueba satisfactoria.

Prueba de aceptación para la historia de usuario Gestionar Proyecto.

Caso de Prueba de Aceptación

Código: HU3_P9

Historia de Usuario: Gestionar Proyecto

Nombre: Devolver la fecha en que fue registrado en el sistema un proyecto determinado.

Descripción: Probar que se devuelva la fecha en que fue registrado en el sistema un proyecto determinado en la interfaz.

Condiciones de Ejecución:

La aplicación debe ser ejecutada correctamente por el Jefe de Polo.

Comprobar la existencia del proyecto en la base de datos.

Entrada / Pasos de ejecución: Se intenta mostrar la fecha en que fue registrado en el sistema un proyecto determinado en la interfaz.

Resultado Esperado: Mostrar la fecha en que fue registrado en el sistema un proyecto determinado correctamente.

Evaluación de la Prueba: Prueba satisfactoria.

Prueba de aceptación para la historia de usuario Gestionar Proyecto.

Caso de Prueba de Aceptación	
Código: HU3_P10	Historia de Usuario: Gestionar Proyecto
Nombre: Devolver los miembros que no están en proyecto.	
Descripción: Probar que se muestren los miembros que no están en proyecto.	
Condiciones de Ejecución: La aplicación debe ser ejecutada correctamente por el Jefe de Polo. Comprobar la existencia de miembros en la base de datos.	
Entrada / Pasos de ejecución: Se intenta mostrar los miembros que no están en proyecto.	
Resultado Esperado: Mostrar los miembros que no están en proyecto.	
Evaluación de la Prueba: Prueba satisfactoria.	

4.6 CONCLUSIONES

En este capítulo se aplicaron las tarjetas CRC (Contenido, responsabilidad y colaboración) para el diseño de la aplicación con el objetivo de hacer entendible las funcionalidades del mismo. Se muestra el modelo de datos de la aplicación, también se trabaja en las fases de implementación y prueba desarrollando las tareas correspondientes para dar solución a las historias de usuarios y las pruebas de aceptación que propician al cliente conformidad y seguridad ante el sistema. Con el fin de este capítulo se da por terminada la propuesta que trae este trabajo.

CONCLUSIONES

Durante el desarrollo de esta aplicación se realizó un estudio que permitió tener un conocimiento de la situación actual y las tendencias de los sistemas de gestión de información en el país. Se demostró la necesidad de desarrollar un sistema que fuese capaz de centralizar la información del Polo de Software Educativo y Multimedia para lograr una mayor organización y control de los volúmenes de información y en las actividades realizadas dentro del Polo así como lograr un análisis integral de los procesos con menor complejidad y menor consumo de tiempo a las personas relacionadas con este trabajo. Se presenta una aplicación Web que permite gestionar todo lo referente a cursos, diplomados, maestrías, eventos, tesis, investigaciones y proyectos, brindando así al jefe de Polo y a los jefes de área un punto de acceso donde este puede obtener toda la información necesaria, así como obtener repostes de las diferentes área de forma más eficiente cumpliendo con sus expectativas, demás permitiendo la constante actualización de todo su contenido.

El desarrollo del trabajo utilizando la metodología XP permitió documentar el mismo desde el comienzo, lo que facilitará su estudio por parte de futuros desarrolladores, permitiendo de esta forma, una comprensión más rápida y fácil de la concepción general del sistema. La versión del sistema obtenida, constituye una base para el trabajo del equipo, así como para futuras versiones del sistema.

RECOMENDACIONES

Como resultado del proceso de investigación y realización de la aplicación han surgido ideas que serían recomendables tener en cuenta para un futuro perfeccionamiento del sistema, a continuación se listan las mismas:

1. Agregar nuevas funcionalidades al sistema de acuerdo con las expectativas y las necesidades existentes dentro del Polo de Software Educativo y Multimedia.
2. Continuar con la línea de desarrollo del sistema.

REFERENCIAS BIBLIOGRÁFICAS

1. **ESTRATEGIA Y GESTIÓN DE LA INFORMACIÓN** [En línea] [Citado el: 19 de Diciembre de 2008.]
<http://www.gestiopolis.com/administracion-estrategia/sistemas-de-gestion-de-informacion-en-estudio-de-medio-ambiente.htm>.
2. **LA FORMACION ANTE LAS TENDENCIAS ACTUALES** [En línea] [Citado el: 20 de Diciembre de 2008.]
<http://elprofesionaldelainformacion.metapress.com/app/home/contribution.asp?referrer=parent&backto=issue,2,13;journal,43,62;linkingpublicationresults,1:105302,1>.
3. **QUE ES SIGCO** [En línea] [Citado el: 5 de Enero de 2009.]
<http://www.pnud.bo/sigco/queessigco.htm>.
4. **DIGITALLEARNING** [En línea] [Citado el: 7 de Enero de 2009.]
<http://digitallearning.es/>.
5. **PORTAL DE VILLA CLARA** [En línea] [Citado el: 20 de Enero de 2009.]
<http://www.idict.villaclara.cu/productos-servicios/cartera-productos-servicios/consultorias/gestion-informacion/>.
6. **LENGUAJE DE PROGRAMACIÓN** [En línea] [Citado el: 30 de Enero de 2009.]
<http://es.kioskea.net/contents/langages/langages.php3>
7. **JAVA SCRIPT** [En línea] [Citado el: 30 de Enero de 2009.]
https://developer.mozilla.org/index.php?title=Es/Gu%C3%ADa_JavaScript_1.5/Concepto_de_JavaScript
8. **LENGUAJE DE PROGRAMACIÓN HTML** [En línea] [Citado el: 1 de Febrero de 2009.] www.monografias.com/trabajos7/html/html.shtml.
9. **Fahnle, Pablo. ASP.NET** [En línea] [Citado el: 4 de Febrero de 2009.]
www.programacion.com/asp/articulo/aspnet_queues/.
10. **Alvarez, Miguel Ange. JAVA** [En línea] [Citado el: 5 de Febrero de 2009.]
www.desarrolloweb.com/articulos/497.php.
11. **MYSQL** [En línea] [Citado el: 6 de Febrero de 2009.]
<http://www.webestilo.com/mysql/intro.phtml>
12. **QUÉ ES MYSQL** [En línea] [Citado el: 7 de Febrero de 2009.]
http://www.netpecos.org/docs/mysql_postgres/x57.html.
13. **MICROSOFT SQL SERVER** [En línea] [Citado el: 7 de Febrero de 2009.]
es.wikipedia.org/wiki/Microsoft_SQL_Server.
14. **ORACLE IBERIA** [En línea] [Citado el: 8 de Febrero de 2009.]
www.oracle.com/global/es/index.html.

15. **METODOLOGÍA DE DESARROLLO DE SOFTWARE** [En línea] [Citado el: 9 de Febrero de 2009.]
www.wikilearning.com/.../metodologias_de_desarrollo_de_software-
16. **METODOLOGÍA RUP** [En línea] [Citado el: 8 de Febrero de 2009.]
<http://teleformacion.uci.cu/mod/resource/view.php?id=2241>.
17. **METODOLOGÍA EXTREME PROGRAMMING** [En línea] [Citado el: 9 de Febrero de 2009.]
www.informatizate.net/articulos/metodologias_de_desarrollo_de_software_07062004.
18. **Gómez, Diego. LO MEJOR DE XP** [En línea] [Citado el: 10 de Febrero de 2009.]
www.dosideas.com/metodologias/546-lo-mejor-de-xp.html.
19. **DEFINICIÓN DE SERVIDOR WEB** [En línea] [Citado el: 11 de Febrero de 2009.]
<http://www.masadelante.com/faqs/servidor-web>
20. **APACHE** [En línea] [Citado el: 12 de Febrero de 2009.]
www.apache.org/.
21. **SERVICIOS DE INTERNET INFORMATION SERVER** [En línea] [Citado el: 12 de Febrero de 2009.]
www.cristalab.com/tutoriales/instalar-apache-internet-information-server-iis-y-mysql-en-windows.
22. **DREAMWEAVER** [En línea] [Citado el: 13 de Febrero de 2009.]
www.adobe.com/es/products/dreamweaver/.
23. **EMS SQL MANAGER 2005** [En línea] [Citado el: 16 de Febrero de 2009.]
www.freedownloadcenter.com/es/Negocio/Aplicaciones/EMS_SQL_Manager_2005_for_SQL_Server.html.
24. **ZEND STUDIO** [En línea] [Citado el: 16 de Febrero de 2009.]
es.wikipedia.org/wiki/Zend_Studio.

BIBLIOGRAFÍA

1. [En línea] www.idict.villaclara.cu/productos-servicios/cartera-productos-servicios/consultorias/gestion-informacion/.
2. [En línea] es.wikipedia.org
3. [En línea] www.gestiopolis.com/administracion.../sistemas-de-gestion-de-informacion-en-estudio-de-medio-ambiente.htm.
4. [En línea] www.tendenciasactuales.com.
5. [En línea] www.oit.org/public/spanish/region/ampro/cinterfor/publ/sala/vargas/for_comp/i.html.
6. [En línea] mx.geocities.com/el_inge5000/Tendenciasactuales.html.
7. [En línea] www.reliefweb.int/SIGCO/loprimeroenemergencia.html.
8. [En línea] insma.udg.es/ambientalitzacio/web_alfastinas/publicacio/10%20Pinar.pdf
9. [En línea] digitallearning.es/.
10. [En línea] www.idict.villaclara.cu/productos-servicios/cartera-productos-servicios/consultorias/gestion-informacion/.
11. [En línea] msguayaquil.com/forums/p/1033/2102.aspx.
12. [En línea] www.debugmodeon.com/item/249640/el-patron-mvc.
13. [En línea] caraballomaestre.blogspot.com/2009/02/arquitectura-j2ee-patron-mvc.html.
14. [En línea] www.mitecnologico.com/Main/LenguajesProgramacionDelLadoDelCliente.
15. [En línea] zenkius.blogspot.com/2008/03/tecnologas-del-lado-del-cliente.html.
16. [En línea] www.javascript.com/.
17. [En línea] es.geocities.com/hipertextomarcadolenguaje/.
18. [En línea] www.webestilo.com/css/.
19. [En línea] www.sidar.org/recur/desdi/mcss/index.php.
20. [En línea] www.adelat.org/media/docum/nuke_publico/lenguajes_del_lado_servidor_o_cliente.html.
21. [En línea] www.maestrosdelweb.com/principiantes/los-diferentes-lenguajes-de-programacion-para-la-web/.
22. [En línea] www.silicontower.net/alojamiento-web/caracteristicas-detalladas/lenguajes-programacion.
23. [En línea] www.bitpipe.com/tlist/PHP.html.
24. [En línea] [triana.escet.urjc.es/apliweb/URJC-AW-PHP-08-09.pdf](http:// triana.escet.urjc.es/apliweb/URJC-AW-PHP-08-09.pdf).

25. [En línea] www.asp.net/ES/.
26. [En línea] www.java.com/es/.
27. [En línea] www.error500.net/garbagecollector/archives/categorias/bases_de_datos/sistema_gestor_de_base_de_datos_sgb.
28. [En línea] www.monografias.com/trabajos29/comparacion-sistemas/comparacion-sistemas.shtml.
29. [En línea] dev.mysql.com/doc/refman/5.0/es/index.html.
30. [En línea] www.mysql-hispano.org/.
31. [En línea] www.desarrolloweb.com/directorio/bases_de_datos/mysql/.
32. [En línea] www.oracle.com/global/es/index.html.
33. [En línea] www.oracle.com/global/es/products/database/index.html.
34. [En línea] ww.informatizate.net/articulos/metodologias_de_desarrollo_de_software_07062004.html.
35. [En línea] www.slideshare.net/fandresvm/metodologia-para-el-desarrollo-del-software
36. [En línea] www.rational.com.ar/herramientas/rup.html.
37. [En línea] <https://pid.dsic.upv.es/C1/Material/Documentos%20Disponibles/Introducción%20a%20RUP.doc>.
38. [En línea] dialnet.unirioja.es/servlet/articulo?codigo=2821750.
39. [En línea] brconsulting.info/portal/articulos/metodologias-de-desarrollo/extreme-programming---xp.html .
40. [En línea] www.masadelante.com/faq-servidor-web.
41. [En línea] www.infor.uva.es/~jvegas/cursos/buendia/pordocente/node20.html
42. [En línea] www.desarrolloweb.com/manuales/36/.
43. [En línea] www.adobe.com/es/products/dreamweaver/.
44. [En línea] www.desarrolloweb.com/articulos/332.php.
45. [En línea] www.bajame.net/EMS-SQL-Manager-2005-for-PostgreSQL.
46. [En línea] www.tufuncion.com/zend-studio.

ANEXOS• **Anexo 1**Descripción de una historia de usuario.

Historia de Usuario	
Número de la historia de usuario.	Nombre de la historia de usuario.
Personas que se relacionan con ésta historia de usuario.	
La prioridad a la hora de la implementación.	Riesgo o complejidad para implementar esta historia de usuario.
Tiempo que demorará la implementación de esta historia de usuario, estimado en semanas	Número de la iteración a la cual pertenece esta historia de usuario.
Programadores responsables de la implementación de esta historia de usuario.	
Breve descripción de esta historia de usuario.	
Observaciones de la historia de usuario.	

• **Anexo 2**Descripción de una Tarjeta CRC

<u>Nombre de la Clase</u>	
Funcionalidades de la Clase	Colaboraciones con otras Clases
Se orientan todas las funcionalidades que debe realizar esta clase.	Se ponen los nombres de las clases que colaboran o se relacionan con ésta.

• **Anexo 3**Descripción de una tarea asignada a una historia de usuario.

Tarea	
Número de la tarea.	Número de la historia de usuario a la cual pertenece la tarea.
Nombre de la tarea a realizar.	

Tipo de tarea, de configuración o de desarrollo.	Tiempo que demorará la realización de ésta tarea en semanas.
Fecha en que se comienza a trabajar en ésta tarea.	Fecha en que se culminará el trabajo de ésta tarea.
Programadores responsables para la implementación de ésta tarea.	
Breve descripción de la tarea a implementar.	

- **Anexo 4**

Descripción de una prueba de aceptación para una historia de usuario.

Caso de Prueba de Aceptación	
Se pone el número de la historia de usuario y el número de la prueba.	Nombre de la historia de usuario a la cual pertenece ésta prueba.
Nombre de la prueba.	
Breve descripción de que se hará en ésta prueba.	
Condiciones de ejecución para realizar la prueba.	
La entrada o paso de ejecución, es lo que se piensa hacer.	
Resultado esperado en cuanto se culmine la prueba.	
Evaluación de la prueba	

GLOSARIO DE TÉRMINOS

SGI: Sistema de gestión de la información.

SIGCO: Sistema de Gestión de Información sobre Cooperación Internacional. Es un sistema integral de gestión de la información para el manejo de programas desarrollados.

SL: Digital Learning. Es una consultora de formación especializada en el área de las Tecnologías de la Información y la Comunicación.

IDICT: Es un servicio basado en la necesidad de un reposicionamiento de la información como base del conocimiento, que constituye el eje central del cambio hacia una nueva sociedad con empresas perfeccionadas.

MVC: Modelo Vista Controlador. Es un patrón de arquitectura de software que separa los datos de una aplicación, la interfaz de usuario, y la lógica de control en tres componentes distintos.

HTML: Hipertexto Markup Language. Es un Lenguaje de marcas de Hipertexto utilizado normalmente en la World Wide Web.

CSS: Las hojas de estilo en cascada es un lenguaje formal usado para definir la presentación de un documento estructurado, escrito en HTML o XML.

Open Source: Es una tendencia internacional del desarrollo de software que profesa la distribución del código junto a las aplicaciones, se rigen por licencias tales como GNU/GPL.

SGBD: Sistemas Gestores de Bases de Datos. Son un tipo de software que sirven de interfaz entre la base de datos, el usuario y las aplicaciones que la utilizan.

SGBDR: Sistemas Gestores de Bases de Datos Relacionales.

SQL: Lenguaje de consulta estructurado. Es un lenguaje declarativo de acceso a bases de datos relacionales que permite especificar diversos tipos de operaciones sobre las mismas.

OLTP: Procesamiento de Transacciones. Es un tipo de sistemas que facilitan y administran aplicaciones transaccionales, usualmente para entrada de datos y recuperación y procesamiento de transacciones.

RUP: Rational Unified Process. Es una *metodología robusta* que puede ser adaptado a proyectos de mayor o menor complejidad, aplicable a diferentes esferas y ajustable a las necesidades de cada organización.

XP: Extreme Programming. Es una metodología ágil centrada en potenciar las relaciones interpersonales como clave para el éxito en el desarrollo de software.

HTTP: Protocolo de transferencia de hipertexto. Es el protocolo usado para intercambiar archivos (texto, gráfica, imágenes, sonido, video y otros archivos multimedia) en la World Wide Web.

TCP/IP: Es un conjunto de protocolo de red en la que se basa internet y que permiten la transmisión de datos entre redes de computadoras.

LDAP: Lightweight Directory Access Protocol. Es un protocolo a nivel de aplicación que permite el acceso a un servicio de directorio ordenado y distribuido para buscar diversa información en un entorno de red.

FreeBSD: Es un sistema operativo libre para computadoras basado en las CPU de arquitectura Intel.

AOLServer: Es un servidor web de código abierto.

FTP: Es un protocolo de red para la transferencia de archivos entre sistemas conectados a una red TCP, basado en la arquitectura cliente-servidor.

SMTP: Protocolo Simple de Transferencia de correo. Protocolo de red basado en texto utilizado para el intercambio de mensajes de correo electrónico entre computadoras u otros dispositivos.

JSP: Es una tecnología Java que permite generar contenido dinámico para web, en forma de documentos HTML, XML o de otro tipo.

IDE: Es un programa compuesto por un conjunto de herramientas para un programador.

HU: Historia de Usuario.

UCI: Universidad de las Ciencias Informáticas.