



**Universidad de las Ciencias Informáticas**

**Facultad 2**

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

**LANGLISH: SISTEMA DE APOYO A LA  
AUTOGESTIÓN DEL CONOCIMIENTO EN LOS  
CASIE**

**Autoras:** Dariela Ferreiro Cabrera  
Elayne Yinet Elejalde Román

**Tutores:** Dr. C. Edistio Yoel Verdecia Martínez  
Msc. Karina Sánchez Tamayo

**Consultante:** Msc. Maura Berta Hidalgo

**La Habana**

**Junio del 2012**

**Declaración de Autoría**

Declaramos ser autores de la presente tesis y reconocemos a la Universidad de las Ciencias Informáticas los derechos patrimoniales de la misma, con carácter exclusivo. Para que así conste firmamos la presente a los \_\_\_\_ días del mes de \_\_\_\_\_ del año \_\_\_\_\_.

Elayne Yinet Elejalde Roman

Dariela Ferreiro Cabrera

---

Firma del autor

---

Firma del autor

Dr.C. Edistio Yoel Verdecia Martínez

Msc. Karina Sánchez Tamayo

---

Firma del Tutor

---

Firma del Tutor

## **Dedicatoria**

De: Dariela Ferreiro Cabrera.

A mi mamá Bertha Rosa Cabrera Carbonell, la persona más buena que he conocido y una excelente madre.

A mi hermana Danay Ferreiro Cabrera por ser la mejor hermana del mundo y consentirme tanto.

A mis abuelos Berta y Raúl que son mis segundos padres y de quienes he recibido inigualable apoyo y ternura.

A mi papá Alexey Ferreiro Suárez, cuya voluntad siempre fue que yo estudiara y llegara donde estoy hoy.

A mi tía Maricela Cabrera por ser la más responsable de la familia y cuidarnos tanto.

A mi novio Frank Hector Rios Morales por su compañía y amor estos años.

De: Elayne Yinet Elejalde Roman

A mi abuelita Margarita que donde quiera que esté en estos momentos, sé que está muy orgullosa de mí.

A mi mamá que con todo su amor y cariño me formó para lograr ser la persona que soy ahora.

A mis tíos Chichito y Dago, por ser los padres que siempre me cuidaron y creyeron ciegamente en mí.

A los hombres más importantes de mi vida, que a pesar de ser jóvenes siempre tuvieron un tiempo para aconsejarme y guiarme en mis decisiones, mi primo Osgel y mi hermano Handy.

## **Agradecimientos**

Queremos agradecer en general a todos aquellos que de una forma u otra han contribuido con el desarrollo de este trabajo de diploma:

A nuestros padres.

A nuestros hermanos.

A nuestros tutores.

A nuestros amigos.

GRACIAS por contribuir a que este sueño se hiciera realidad.

## **Resumen**

En el presente trabajo se describe la construcción de una solución informática que favorece la autogestión del conocimiento en el idioma inglés y propicia su uso en el proceso de enseñanza aprendizaje (PEA) de las asignaturas del idioma extranjero desde los Centros de Auto-Aprendizaje y Servicios de Idiomas Extranjeros (CASIE) de la Universidad de las Ciencias Informáticas (UCI) aprovechando las ventajas que ofrecen las Tecnologías de la Información y las Comunicaciones (TIC).

La solución construida en un ambiente web, bajo el marco de trabajo Symfony y PostgreSQL como gestor de base de datos, siguiendo la metodología ágil XP para su desarrollo, permite la realización de diagnósticos a los estudiantes considerando su año, y el tratamiento diferenciado teniendo en cuenta las asignaturas que le imparten y el nivel de autogestión dado según el diagnóstico.

Con el fin de lograr este propósito, se realizó un estudio de los sistemas automatizados de gestión del conocimiento y los existentes vinculados al campo de acción, así como las definiciones de términos utilizados en la realización del sistema. Además se describen las funcionalidades que debe cumplir la aplicación, modelándose además todo el proceso de construcción de la misma.

Palabras claves: autogestión del conocimiento, asignaturas de idiomas extranjeros, sistema de apoyo.

---

---

## ÍNDICE

INTRODUCCIÓN.....	8
CAPÍTULO I: AUTOGESTIÓN DEL CONOCIMIENTO. SISTEMAS UTILIZADOS PARA FOMENTAR LA GESTIÓN DEL CONOCIMIENTO .....	14
1.1 Autogestión del conocimiento desde la enseñanza del idioma inglés.....	14
1.2 Sistemas automatizados de gestión del conocimiento.....	18
1.3 Herramientas y metodologías de los sistemas web .....	21
1.4 Conclusiones parciales.....	32
CAPÍTULO II: EXPLORACIÓN Y PLANIFICACIÓN. ....	33
2.1 Problema y situación problemática .....	33
2.2 Objeto de automatización .....	33
2.3 Descripción del sistema .....	34
2.4 Exploración del Sistema .....	35
2.5 Selección de metodologías, tecnologías y herramientas .....	40
2.6 Descripción de la Arquitectura .....	42
2.7 Planificación.....	48
2.8 Conclusiones parciales.....	50
CAPÍTULO III: ITERACIÓN Y PRODUCCIÓN.....	52
3.1 Tareas de Ingeniería.....	52
3.2 Tarjetas Clase – Responsabilidad – Colaborador.....	54
3.3 Modelo de Datos.....	56
3.4 Modelo de Despliegue .....	56
3.5 Validación del Sistema.....	57
3.6 Conclusiones parciales.....	71
CONCLUSIONES GENERALES .....	72
RECOMENDACIONES.....	73
GLOSARIO DE TÉRMINOS .....	74
REFERENCIAS BIBLIOGRÁFICAS.....	75
BIBLIOGRAFÍA.....	79
ANEXOS.....	<b>¡ERROR! MARCADOR NO DEFINIDO.</b>
ANEXO1: IDIOMAS MÁS UTILIZADOS EN INTERNET.....	<b>¡ERROR! MARCADOR NO DEFINIDO.</b>

---

ANEXO2: ENCUESTAS REALIZADAS A ESTUDIANTES Y PROFESORES SOBRE EXPERIENCIAS EN EL EVE/A CON LA ASIGNATURA INGLÉS.¡ERROR! MARCADOR NO DEFINIDO.

ANEXO3: HISTORIAS DE USUARIO..... ¡ERROR! MARCADOR NO DEFINIDO.

ANEXO4: TAREAS DE INGENIERÍA. .... ¡ERROR! MARCADOR NO DEFINIDO.

ANEXO5: TARJETAS CRC DEL SISTEMA..... ¡ERROR! MARCADOR NO DEFINIDO.

ANEXO6: CASOS DE PRUEBA DE ACEPTACIÓN. .... ¡ERROR! MARCADOR NO DEFINIDO.

ANEXO7: RESULTADOS DE LAS PRUEBAS DE ACEPTACIÓN.¡ERROR! MARCADOR NO DEFINIDO.

---

## Introducción

El mundo de hoy ha alcanzado un enorme desarrollo en todas las esferas de la vida, para lo que interactúa constantemente con nueva información y conocimientos; el desarrollo de la informática surgió sin duda respondiendo a las necesidades de desarrollarse que tiene el hombre. La informática ha traído consigo cambios a nivel mundial, permitiendo compartir la información y los avances tecnológicos, eliminando barreras de tiempo y espacio. Este acceso a la información ha llegado también a los sistemas educativos, el uso de tecnologías para la educación ha roto con los esquemas tradicionales de enseñanza brindando herramientas apropiados para el desarrollo del intelecto e interacción de los estudiantes.

Según Reportes elaborados por European SchoolNet en el marco de trabajo de la Comisión Europea del sector de las Tecnologías de la Información y la Comunicación (TIC):

- Las TIC tienen un impacto positivo en el desempeño educativo en las escuelas de primaria, en especial en Inglés, menor en ciencias, y no impactan de igual manera las matemáticas.
- En los países OECD (Organización para la Cooperación y el Desarrollo Económico) hay una asociación positiva entre el tiempo dedicado al uso de las TIC y el desempeño de los estudiantes en las pruebas PISA (Programa Internacional para la Evaluación de Estudiantes).
- Las escuelas con mayor madurez digital (e-maturity) muestran un incremento más rápido en los puntajes de desempeño si se comparan con las que tienen niveles inferiores.
- Las escuelas con buenos recursos en TIC alcanzan mejores resultados que aquellas que tienen una dotación pobre.
- La inversión en las TIC impacta los estándares educativos y lo hace de mayor manera cuando existe un terreno fértil en las escuelas para hacer uso efectivo de éstas.
- La introducción de tableros (pizarras) digitales interactivos mejora los resultados de los estudiantes en las pruebas nacionales de Inglés (especialmente en

---

alumnos con bajo rendimiento en escritura), Matemáticas y Ciencias, si se los compara con los estudiantes de escuelas que no cuentan con estos.

Cuba también ha desarrollado planes para el desarrollo de la Informática, pues esta trae consigo avances en el sistema político, económico y social del país.

La utilización de las TIC en el sector educacional cubano ha permitido alcanzar varios resultados, como se pueden mencionar:

- Reducción del número de alumnos por aulas y por consecuencia el aumento de estas.
- Desarrollo del Programa de Profesores Generales Integrales.
- Homogenización de los métodos de enseñanza.
- Garantizar una mayor calidad en las “teleclases”.(1)

A pesar de las limitaciones económicas, Cuba puede exponer resultados significativos en esta rama y que van encaminados al beneficio de la sociedad cubana. A continuación se muestran algunos ejemplos:

- Se creó la Oficina Nacional para la Informatización, de alcance nacional y que tiene como principal objetivo, promover el proceso del uso masivo, ordenado, eficiente y eficaz de la Informática en Cuba.
- Como ejemplo de la información especializada y el aprovechamiento eficiente de los servicios disponibles en la Red se encuentra INFOMED (Portal de Salud Cubano), con servicios de Universidad Virtual, Biblioteca Virtual, Red de Telemedicina, Acceso a Bases de Datos Especializadas y correo electrónico.
- Todos los centros de enseñanza del país usan las TIC como apoyo a los programas de clases, junto a otras tecnologías como la televisión.
- Se crearon las condiciones de infraestructura necesarias para conectar a la red los Institutos Superiores Pedagógicos del país, y a estos con sus sedes municipales.
- La Universidad de las Ciencias Informáticas (UCI), creada en el 2002 prepara fuerza de trabajo calificada para desarrollar esencialmente la industria del software y la informatización de la sociedad.
- Se han creado y utilizado softwares educativos en los diferentes niveles de enseñanza. Se cuenta con tres colecciones de gran estimación para maestros y alumnos: “Multi-saber”, —primer gran esfuerzo nacional dedicado a la Enseñanza

---

Primaria—; “El Navegante”, para la Secundaria Básica; y “Futuro”, diseñado para el Bachillerato, la Enseñanza Técnica-Profesional y la Educación de Adultos. (2)

Un avance significativo en la descentralización de la información trajo consigo la creación y uso de Internet como una herramienta de comunicación y almacenamiento de información mundial sin comparación, información que está en su mayoría en idioma inglés, como se muestra en el Anexo1 de este documento.

A partir de lo anterior, el estudio del idioma inglés adquiere gran importancia, su conocimiento permite el intercambio de la información entre personas de diversas culturas, promoviendo la cooperación entre los países e instituciones.

A través de los años se han ido perfeccionando los métodos utilizados en el proceso de enseñanza-aprendizaje del idioma inglés, con el objetivo de obtener mejores resultados. Se reconoce que para los especialistas de la rama de la informática y la computación el estudio de este idioma se hace indispensable para mantenerse actualizado. De ahí la necesidad de que los especialistas de este campo estén preparados no sólo en el aspecto técnico sino también en este idioma, ya que sin ello, muchos no podrán siquiera llegar a demostrar su formación técnica, al no estar capacitados para superar lo que en muchos casos es la primera barrera, la del idioma.

La UCI tiene entre sus propósitos que los estudiantes utilicen los medios de enseñanza-aprendizaje más provechosos para su formación. En esta universidad el estudio del idioma inglés constituye parte del currículo de los estudiantes, demostrándose su uso no solo en esta disciplina sino en las de carácter técnico. Además, en la carrera de Ingeniería en Ciencias informáticas una de las competencias genéricas definidas en su modelo del profesional es la “Capacidad de comunicación en un segundo idioma”, al ser el idioma inglés uno de los más difundidos y de mayor importancia en el mundo científico, y en especial para la Computación y la Informática, es el que se incluye como parte de los estudios.

Una de las estrategias de la UCI para fortalecer el estudio del idioma inglés, es la utilización de los Centros de Auto-Aprendizaje y Servicios de Idiomas Extranjeros (CASIE), estos centros se han equipado con los recursos tecnológicos necesarios para fortalecer el aprendizaje de los idiomas extranjeros. Sin embargo los CASIE no son explotados en todas sus potencialidades. Se ha observado que los estudiantes al visitar los CASIE para usar sus servicios se ven obligados a contar con el asesor del centro para que los guíe en su actividad y muchas veces tienen que esperar porque este se encuentra

---

atendiendo a otro visitante, por lo que el aprovechamiento de estos centros por parte de los estudiantes es insuficiente, y los recursos tanto físicos como virtuales no garantizan del todo la autopreparación para la gestión del conocimiento en inglés del futuro profesional.

Una de las deficiencias actuales de los CASIE es que no cuentan con aplicaciones que guíen y fomenten el interés del alumnado a visitar el centro con fines de aprendizaje y el incremento gradual de la gestión del conocimiento, lo que sería satisfactorio para elevar el nivel de uso de los CASIE y en aras de preparar mejor al estudiantado en el idioma inglés.

Si bien es importante la utilización de herramientas telemáticas, que permiten fusionar la telecomunicación y la informática, y de softwares educativos diseñados con calidad, no menos relevancia posee la gestión de la información y su comprensión. “Las TIC dinamizan la transmisión de la información, pero la gestión del conocimiento eficaz depende de la cognición humana y de la comunicación” (3).

En la UCI se utiliza un Entorno Virtual de Aprendizaje (EVA), implementado sobre la plataforma Moodle, como herramienta de apoyo en el proceso de enseñanza aprendizaje (PEA) de la universidad, el cual ofrece apoyo a las asignaturas de la disciplina Idioma Extranjero. Para comprobar la eficacia de su uso en el tratamiento de la asignatura Idioma Inglés fueron realizadas encuestas, por la profesora de la Facultad 2 Maura Berta Hidalgo Díaz, a una muestra de estudiantes de segundo y cuarto año sobre sus experiencias de aprendizaje en el idioma inglés utilizando el EVA (Ver Anexo2). A partir del procesamiento de los resultados de las encuestas se ha observado que los estudiantes:

1. Presentan dificultades en la comprensión de la información y muestran bajos niveles de independencia cognoscitiva.
2. Son inadecuadas las estrategias de lectura en idioma inglés, por consiguiente, no son capaces de autogestionar la información contenida en los materiales publicados en el entorno virtual en el idioma, solo preparándose en lo mínimo indispensable que necesitan en la asignatura inglés.
3. La gestión del conocimiento en el idioma inglés enfocadas en las disciplinas de la especialidad es insuficiente, lo que limita el éxito en su formación académica y como futuro profesional de la informática.

Los resultados arrojados por dichas encuestas evidencian que se hace necesario en la universidad continuar buscando alternativas para alcanzar un nivel adecuado de gestión, comprensión y aplicación del conocimiento en el idioma Inglés por sus estudiantes.

---

anterior puede ser llevado a cabo mediante los recursos que posee la UCI, entre ellos los CASIE, estos centros poseen los medios necesarios para ayudar en la mejora del PEA de las asignaturas de Idioma Extranjero.

Para resolver las insuficiencias que se están presentando con el EVA y los CASIE en la UCI, sería necesario un sistema que vinculara la gestión de la información con la comprensión de la información y la aplicación de la misma; conjunto al esfuerzo de los asesores de la asignatura de inglés, los profesores de otras materias y los estudiantes.

Para darle solución a la **situación problemática** planteada se puede definir el siguiente **problema a resolver**: ¿Cómo propiciar la autogestión del conocimiento en el PEA de las asignaturas del idioma inglés desde los CASIE en la Universidad de las Ciencias Informáticas aprovechando las ventajas que ofrecen las TIC?

Y el **objeto de estudio** lo constituye: el proceso de autogestión del conocimiento en el PEA de las asignaturas del idioma inglés, enmarcándose en el **campo de acción**: la autogestión del conocimiento en el PEA de las asignaturas del idioma inglés desde los CASIE utilizando las TIC.

Se plantea como **idea a defender**: Con el desarrollo de un sistema de apoyo a la autogestión del conocimiento para los CASIE se contará con una herramienta que puede favorecer el PEA de las asignaturas del idioma inglés en la UCI aprovechando las ventajas que ofrecen las TIC.

Se define como **objetivo general**: Desarrollar un sistema de apoyo para los CASIE en la UCI que favorezca la autogestión del conocimiento en el PEA de las asignaturas del idioma inglés aprovechando las ventajas que ofrecen las TIC.

Como **tareas de investigación** se identificaron:

1. Caracterizar la utilización de las TIC en la enseñanza del idioma Inglés en el contexto internacional, nacional y local.
2. Caracterizar las tendencias y tecnologías actuales para el desarrollo de los sistemas utilizados para la Gestión del Conocimiento.
3. Realizar el análisis y diseño del sistema informático para el apoyo a la autogestión del conocimiento en los CASIE.
4. Implementar las funcionalidades de la primera etapa.
5. Validar el sistema informático desarrollado.

---

Para afrontar el problema, cumplir con los objetivos y las tareas de la investigación se utilizaron los siguientes métodos de la investigación:

### **Métodos Teóricos**

Analítico-Sintético, donde se realizó la lectura de diferentes documentos para entender las peticiones del cliente, a partir del análisis del negocio, funcionalidades o requisitos, y para formular conclusiones o reelaborar ideas a través de la investigación realizada.

Histórico-Lógico permitió estudiar el estado actual de los sistemas de gestión de conocimiento a partir del análisis de su evolución y las etapas principales por las que han transitado.

### **Métodos Empíricos:**

Análisis documental, donde se realizó un estudio y revisión de la documentación especializada relacionada con la utilización de las TIC en la enseñanza de idioma inglés, adquiriendo así la información necesaria que permitió realizar el proceso de investigación.

Este trabajo se estructura en una introducción, tres capítulos, conclusiones, recomendaciones, bibliografía y un cuerpo de anexos que complementan los capítulos presentados.

El primer capítulo comprende el estudio de los principales conceptos vinculados a la investigación, así como el estudio de las principales tendencias, técnicas, tecnologías, metodologías, sistemas y otros elementos del desarrollo de software. En el capítulo dos se abordan los procesos a automatizar, la descripción general de la propuesta de sistema, como debe funcionar, las metodologías, lenguajes y herramientas a ser usadas en la construcción del software. En el tercer y último capítulo se exponen las tareas necesarias para la implementación de las funcionalidades definidas, el modelo de datos y las características de la arquitectura según las tecnologías seleccionadas. Se explica la necesidad de las pruebas de aceptación para validar el sistema y la confección de los casos de pruebas que se utilizan para realizarlas.

---

## **Capítulo I: Autogestión del Conocimiento. Sistemas utilizados para fomentar la Gestión del Conocimiento**

El presente capítulo tiene como objetivo abordar aspectos y conceptos teóricos que permitan describir el estado del arte del objeto y campos declarados en la investigación, para ello se analizan aspectos teóricos y prácticos de los términos de gestión de la información y gestión del conocimiento como premisas para la auto-gestión del conocimiento en inglés en el contexto de los CASIE en la UCI. Es objetivo además analizar las diferentes herramientas, tecnologías, metodologías de desarrollo de software, que se van a tener en cuenta para poder seleccionar las más adecuadas una vez descritas las características del sistema a desarrollar.

### **1.1 Autogestión del conocimiento desde la enseñanza del idioma inglés**

En la escuela tradicional, la enseñanza se centraba en el maestro como transmisor del conocimiento, mientras que los alumnos se comportaban como receptores pasivos de la información que este les hacía llegar. En las tendencias actuales, la enseñanza tiene como protagonista al estudiante en función de la gestión de la información y del conocimiento. De ahí que, resulta necesario promover vías para que esta gestión sea efectiva.

Como punto de partida del presente epígrafe se puede plantear la siguiente pregunta ¿Qué es el conocimiento?, en la elaboración de su respuesta se consultaron un grupo de documentos para obtener las siguientes definiciones.

El conocimiento se define como “Una verdad justificada” (4). Según Mazo y Ortiz en 1998 (5) cuando plantean que el conocimiento surge cuando una persona considera, interpreta y utiliza la información de manera combinada con su propia experiencia y capacidad.

Existen diversas definiciones sobre gestión del conocimiento, las cuales dependen de su enfoque:

- Según Artilles Visval (2005) “La gestión de conocimiento hace énfasis en facilitar y gestionar actividades relacionadas con el conocimiento tales como la creación, captura, transformación y uso. Su función es planificar, implementar, operar y gestionar todas las actividades relacionadas con el conocimiento y los programas requeridos para la gestión efectiva del capital intelectual” (6).

- 
- Según Andreu y Sieber (1999) “La gestión del conocimiento es el proceso que continuamente asegura el desarrollo y la aplicación de todo tipo de conocimientos pertinentes de una empresa con objeto de mejorar su capacidad de resolución de problemas y así contribuir a la sostenibilidad de sus ventajas competitivas” (7).
  - Según Estrada Sentí y Febles Rodríguez (2002) “Proceso donde se añade valor a los conocimientos existentes y se generan nuevos conocimientos científicos, un nuevo mercado y nuevos servicios” (8)

Actualmente se apoya con mayor énfasis en las instituciones educacionales el propósito de alcanzar la gestión del conocimiento, lo cual resulta más importante que la cantidad de conocimiento almacenado. Se comparte la opinión de Núñez (3) quien plantea que, es necesaria una gestión consciente y planificada del conocimiento que potencie su creación, transferencia, conservación y reutilización en diferentes contextos.

Por todo lo anterior, se puede plantear que la gestión del conocimiento depende de factores tales como la información que se maneje y se orienta hacia el logro de metas según el contexto que se utilice.

De acuerdo con Castillo “el ser humano debe concentrarse más en aprender a trabajar en la adquisición del conocimiento que en la adquisición de la información” (9). Para ello se requiere que los estudiantes posean habilidades en cuanto a la búsqueda, selección, clasificación y procesamiento de la información. Según Prieto (10) este conjunto de habilidades determina el actual escenario de autogestión de la información propiciando las bases para entablar la autogestión del conocimiento.

La **autogestión del conocimiento** es definida por Davenport como “el proceso sistemático de buscar, organizar, filtrar y presentar la información con el objetivo de mejorar la comprensión de las personas en una específica área de interés” (11).

En la UCI, según las encuestas realizadas, las observaciones de las clases y los análisis realizados en las reuniones metodológicas de los profesores, los estudiantes se preparan para aprobar la asignatura Idioma Inglés, pero no hacen uso de esta lengua como instrumento para favorecer la gestión del conocimiento de forma intencionada en las disciplinas de la especialidad, lo que contribuiría al éxito en su formación académica y como futuro profesional de la Informática. Sin la consulta de materiales en idioma inglés, el nivel de actualización de los estudiantes y profesionales es insuficiente, limitando su participación y contribución al desarrollo de la Informática en la sociedad cubana.

---

Cada día se acuña un nuevo término en el campo de la Informática, el cual debe ser decodificado para su uso y socialización antes de su obsolescencia. Por tal motivo, el estudiante debe ser capaz de buscar información en idioma inglés, lograr extraer las ideas esenciales de los textos que consulta y poder resumir el contenido de los mismos para luego poder utilizarlos en función de sus necesidades profesionales.

Debido a la necesidad de proporcionar una cultura para la gestión de la información en idioma inglés durante el período de formación de los estudiantes de la UCI, se analizan las siguientes consideraciones de Núñez Paula (3) para la gestión de la información en una organización, con el propósito de su adaptación con fines académicos.

1. Estudiar las necesidades de información.
2. Buscar, filtrar y seleccionar la información en las fuentes documentales y no documentales del entorno de la organización, e internas.
3. Almacenar organizadamente la información, antes y después de los procesos de filtrado, selección y análisis en diferentes tipos de soportes.
4. Analizar la información en formas diversas con herramientas de diferentes tipos para descubrir en ellas, elementos que le añaden valor y que posibilitan el descubrimiento y aprovechamiento de oportunidades o la solución de problemas.
5. Diseñar y elaborar productos informativos de alto valor agregado en función de los objetivos de la organización y las necesidades que estos generan.
6. Determinar los canales y formas que deben servir de base a los flujos de información en la organización.
7. Garantizar dicho flujo o diseminar la información de modo estable, continuo y oportuno como soporte de los procesos de creación, transferencia y utilización del conocimiento (aprendizaje).

En el ámbito académico la gestión de la información se relaciona con las capacidades, habilidades o estrategias que tributan indirecta o directamente al hecho de aprender a aprender. Para muchos especialistas en el campo de la educación el proceso de búsqueda y procesamiento de la información constituye el centro de atención durante el aprendizaje, pues gestionar la información de manera correcta presupone una adecuada gestión del conocimiento.

---

### 1.1.2 Gestión de la Información

Como se puede apreciar la autogestión del conocimiento no es más que la información procesada y convertida en conocimiento. ¿La información es conocimiento? Según Castillo “La información se asume como una verdad mientras que el conocimiento es una creencia justificada de la verdad ” (9). Otra visión de la información es la dada por Langefors y Monforte. El primero plantea que “la información es cualquier clase de conocimiento o mensaje que puede usarse para mejorar o posibilitar una acción” (12), por lo que se puede deducir que la aplicación de la información puede llegar a tener un valor significativo; el segundo precisa que “la información es un dato o conjunto de ellos que, en un contexto determinado tienen un significado para alguien” (13), interpretándose esta idea se puede plantear que la información solo tendrá sentido para aquel que la utiliza.

La información debe poseer tres características fundamentales: ser completa, ser confiable y ser oportuna (14).

Muchas veces se cuenta con un gran cúmulo de información, sin embargo, es preciso entender que contar con mayor cantidad de información no significa estar mejor informado. Por eso, es muy importante saber seleccionar la información que se posee, y atribuirle importancia y fiabilidad, en dependencia de la fuente que procede. También se deben obtener distintos puntos de vista acerca de la misma información para establecer un criterio sobre ella y poder gestionarla de manera adecuada.

La gestión de información comprende el análisis y procesamiento de la misma para respaldar a las personas en los procesos de toma de decisiones. Según Marrero la gestión de la información es el “proceso que se encarga de suministrar los recursos necesarios para la toma de decisiones, así como para mejorar los procesos, productos y servicios de la organización” (14).

### 1.1.3 Comprensión de la información

La comprensión de la información se basa en la interpretación y procesamiento de las ideas presentes en el texto, lo cual permite determinar desde una perspectiva diferente la idea central, la elaboración de resumen y la proyección de una imagen basada en lo descrito en el texto. En el proceso de comprensión actúan dos tipos de memoria: La memoria a corto plazo y la memoria a largo plazo (15).

---

#### 1.1.4 Aplicación de la información

La aplicación de la información es el empleo de la misma según la problemática dada, dándole una solución o formulando preguntas en busca de una posible respuesta desde un nuevo contexto.

#### 1.2 Sistemas automatizados de gestión del conocimiento

Los sistemas de gestión del conocimiento son herramientas que tienen como objetivo general apoyar los procesos de creación, almacenamiento, recuperación, transferencia y aplicación del conocimiento. En la actualidad las organizaciones han detallado el uso de los sistemas de gestión de conocimiento, destacándose entre ellos:

- Desarrollar una cultura en conocimiento, haciéndolo visible y fomentando conductas que propicien su intercambio.
- Construir una infraestructura de conocimiento. No solo un sistema técnico, sino toda una red de conexiones entre personas, facilitándoles espacio, tiempo y herramientas para que interactúen y colaboren.

Estas herramientas abarcan procesos que podrían incluirse dentro de las funcionalidades de búsqueda o de distribución personalizada de la información, entre ellas se pueden mencionar: los buscadores, los cuales exploran documentos utilizando palabras claves o una información determinada; y los repositorios, que permiten almacenar los datos de una forma estructurada permitiendo realizar posteriores consultas y análisis de la información.

Existen diversos sistemas que apoyan la gestión del conocimiento, desde un simple portal web hasta un sistema de gestión, pero cada cual tiene como característica fundamental, independientemente de las herramientas con que se implementaron, cumplir las metas. Los sistemas y procesos de gestión del conocimiento no son generalizables, por lo que se deben diseñar y desarrollar según el problema real de la organización.

Según David Rodríguez (16) se pueden encontrar algunas limitaciones y dificultades en los sistemas cuando se aplican los procesos de gestión del conocimiento, como son:

- Ausencia de objetivos: Un sistema de ser desarrollado para lograr conseguir los objetivos propuestos por la organización.

- 
- Responsabilidades difusas: Se deben tener clara las responsabilidades que tiene cada usuario en el sistema.
  - Confusión conceptual: se debe esclarecer con qué fin se realiza el sistema, para no hacer un mal uso del mismo.

### **1.2.1 Sistemas existentes vinculados al campo de acción**

En internet se puede encontrar algunos sistemas de gestión del conocimiento vinculados a la educación, de los cuales se pueden mencionar algunos que favorecen el estudio del idioma inglés:

TED-ED: Es una plataforma que contiene diversos videos didácticos de no más de 10 min, estos videos son creados por profesores de todo el mundo en los que se explican, de manera simple, conceptos elementales de materias tan variadas como arte, empresa, economía, diseño, ingeniería, tecnología, salud, ciencia, psicología, lengua, literatura y estudios sociales, también se pueden incluir lecciones para el aprendizaje de los idiomas extranjeros. (17)

LONGAM ENGLISH Interactive: Es un software usado en universidades latinoamericanas, el cual tiene como características: acceso desde cualquier lugar y en cualquier momento, navegación sencilla, permite al profesor evaluar continuamente el progreso de los estudiantes y cuenta además con 8 idiomas diferentes. (18)

Khan Academy: Es un sitio web que tiene como meta principal brindar una educación de alta calidad a cualquiera que lo desee y desde cualquier lugar. Contiene una colección online y gratuita de aproximadamente 2.600 clases mediante videos almacenados en sitios web como Youtube y de ejercicios para enseñar matemática, historia, finanzas, química, biología, astronomía y economía. Los videos en su mayoría se encuentran en inglés, por lo que el sitio cuenta con subtítulos, permitiendo así la práctica del idioma. (19)

De los sistemas antes mencionados se puede destacar que todos se encuentran publicados en internet, siendo esto el primer impedimento para Cuba, pues no todas las personas cuentan con este servicio. Además, Logam English es un software privativo y Khan Academy no puede ser accedido desde Cuba pues entra dentro de los sitios que no le son permitidos a la red cubana por consecuencia del bloqueo impuesto injustamente al país.

---

En la UCI se utilizan algunos softwares para apoyar el aprendizaje de varias de las disciplinas que se imparten en la carrera. La universidad cuenta con un EVA que vincula las diferentes asignaturas que sus estudiantes cursan durante los 5 años de la carrera, entre ellas se encuentran las asignaturas de inglés, en las que los estudiantes realizan tareas de investigación y evalúan los conocimientos adquiridos durante el semestre. Sin embargo el EVA no permite integrar los procesos de diagnóstico y posicionamiento de los estudiantes para darles un tratamiento diferenciado según sus habilidades, por lo que no es capaz de guiar y monitorear el avance que tiene el estudiante durante su aprendizaje ni de reconocer el nivel de conocimiento que posee.

También se desarrollan algunos sistemas a la medida, por ejemplo en la Facultad 3 se encuentra en desarrollo el Tutor Virtual de Evaluación para el Aprendizaje Autónomo de idiomas extranjeros (VIRTEVALL). Este sistema tiene como característica procesar la información a partir de un conjunto de encuestas y cuestionarios realizados al usuario, profundizando en los temas que son de su interés. Los usuarios que utilizan el sistema pueden ser todos los que quieran mejorar la calidad en el proceso de formación académica en esta disciplina. Según los resultados las evaluaciones que alcancen los estudiantes, se podrán ver los estilos de aprendizaje generándose así una ruta de autoaprendizaje a seguir durante el entrenamiento. VIRTEVALL tiene como objetivo lograr que los estudiantes sean autónomos en cuanto a la evaluación del aprendizaje de las lenguas extranjeras y completar una ruta de aprendizaje sin la intervención imprescindible de un profesor.

De manera general se puede concluir que VIRTEVALL es un sistema que facilitará el estudio del idioma inglés de forma independiente.

Los sistemas antes mencionados se pueden catalogar como sistemas de gestión del conocimiento pues de una forma u otra realizan o apoyan la gestión consciente y planificada del mismo por establecer vías para su transferencia, conservación y reutilización. Estos sistemas tienen una meta a cumplir y sus funcionalidades se basan en esos objetivos. Se puede plantear que contribuyen al aprendizaje del inglés, lo cual no garantiza que los instruya para la autogestión del conocimiento del contenido de su especialidad en inglés.

Un sistema de gestión del conocimiento puede propiciar o no, la autogestión. Si se desea lograr la autogestión del conocimiento mediante un software se debe crear para ese fin. Es decir que la información puede ser reflejada o transmitida mediante cualquier medio de

---

información y según la acción que realice el receptor se podrá deducir si solo fue una información recibida o si mediante su proceso se generó conocimiento.

Luego de la investigación realizada se puede plantear que en la UCI no se ha desarrollado aún un software que cubra las necesidades enfocadas a promover la autogestión del conocimiento, es decir, que incluya todas las habilidades de la autogestión de la información y que este a su vez establezca las bases para la autogestión del conocimiento.

### **1.3 Herramientas y metodologías de los sistemas web**

Para el correcto desarrollo de cualquier software es necesario estudiar los aspectos más importantes relacionados con las herramientas, metodologías de desarrollo, lenguajes de programación, y otras tecnologías necesarias que se utilicen con este fin, para analizar y elegir las idóneas a usar en la realización del sistema. Se tiene en cuenta que los software de gestión del conocimiento no poseen un estándar para su implementación pues como se analizó en el epígrafe anterior, su desarrollo depende de las necesidades específicas de la organización, se pueden denominar así tanto portales web como sistemas de gestión, solo debe existir relación entre sus usuarios así como tareas o metas comunes.

#### **1.3.1 Lenguaje de programación**

Los lenguajes de programación facilitan la tarea de programación, ya que disponen de formas que permiten a las personas dar órdenes a las computadoras.

En este trabajo se abordarán algunos lenguajes de programación para la creación de sistemas web, debido a que en la actualidad los softwares de apoyo al aprendizaje son aplicaciones web independientemente de cuál sea su propósito.

Los lenguajes de programación para la web se clasifican en lenguaje del lado del cliente y lenguaje del lado del servidor.

Un **lenguaje del lado cliente** es totalmente independiente del servidor y no necesita tratamientos previos, son aquellos script que se ejecutan en la máquina del cliente y permiten entre otras cosas visualizar la información de manera estructurada. Entre ellos se destacan en la actualidad JavaScript y HTML.

**JavaScript** es un lenguaje de programación orientado a objetos y dinámico, se ejecuta en la parte del cliente. Fue creado por Brendan Eich de la empresa Netscape

---

Communications. Este lenguaje se ejecuta en el navegador, por lo que este último es el encargado de todo el procesamiento. Este lenguaje no necesita de compilación, es multiplataforma y de alto nivel. Es usado fundamentalmente para definir y configurar todo el proceso de interactividad con los usuarios.

Técnicamente, JavaScript es un lenguaje de programación interpretado, por lo que no es necesario compilar los programas para ejecutarlos. En otras palabras, los programas escritos con JavaScript se pueden probar directamente en cualquier navegador sin necesidad de procesos intermedios. (20)

**Hyper Text Markup Language** más conocido como HTML es el lenguaje usado en la World Wide Web, y uno de los más usados actualmente en la gestión de documentos para mostrarlos en hipertexto. Este lenguaje se encarga de controlar en el navegador las imágenes, textos, videos, para lo cual utiliza una serie de etiquetas definidas por el lenguaje.

HTML es un lenguaje de composición de documentos y especificación de ligas de hipertexto que define la sintaxis y coloca instrucciones especiales que no muestra el navegador, aunque si le indica cómo desplegar el contenido del documento, incluyendo texto, imágenes y otros medios soportados. HTML también indica cómo hacer un documento interactivo a través de ligas especiales de hipertexto, las cuales conectan diferentes documentos, ya sea en su computadora o en otras, así como otros recursos de Internet, como FTP y Gopher. (21)

Los **lenguajes del lado del servidor** son aquellos que se generan en el servidor para luego ser enviados al cliente, reciben peticiones de los usuarios y realizan la lógica del negocio para darles respuesta. Entre estos se pueden mencionar PHP y ASP.

**Active Server Pages** (ASP, por sus siglas en inglés) es un lenguaje de scripting creado por Microsoft con el propósito de crear páginas dinámicas desde el servidor. El uso de este lenguaje brinda facilidades para programar sitios web por contar con objetos integrados, pero también tiene la limitante de funcionar solo con el servidor web Internet Information Server (IIS), que ofrece servicios solo para el sistema operativo Microsoft Windows. ASP.NET es parte de la plataforma Microsoft.NET de programación revolucionaria que permite el desarrollo de aplicaciones Web dirigidas a corporaciones. (22)

---

Como lenguaje de programación de ASP.NET se pueden utilizar C# o J#, y para el funcionamiento de las páginas se necesita tener instalado el servidor web Internet Information Services (IIS) y un Framework.Net.

**Hypertext Pre-processor** (PHP, por sus siglas en inglés) es un lenguaje de programación web del lado del servidor utilizado para la creación de páginas web dinámicas, de propósito general. Fue uno de los primeros lenguajes de script empotrados en HTML. Su código es interpretado en el servidor por un procesador PHP resultando en la página web a ser transmitida al cliente. PHP es software libre, bajo la licencia PHP, por tanto no se ve limitado por restricciones o costos.

Durante los últimos años PHP ha crecido hasta ser una de las plataformas web más usadas en el mundo, lista para ser usada en más de tres servidores web desde un lado a otro del globo mundial. Su crecimiento no es solo cuantitativo sino también cualitativo. Cada día más y más compañías confían en PHP para realizar las aplicaciones más importantes dentro de sus negocios. (23)

Uno de los elementos que distingue al desarrollo utilizando PHP es la existencia de múltiples marcos de trabajo (frameworks), la caracterización de algunos de ellos se muestra a continuación. El uso de **frameworks** para PHP permite contar con un grupo de componentes que reducen el tiempo de desarrollo y la complejidad de los sistemas a elaborar, además de proporcionar los elementos necesarios de seguridad y las facilidades para la comunicación con las bases de datos.

Uno de los frameworks más utilizados en la actualidad es **Symfony**, este es un completo framework diseñado para optimizar el desarrollo de las aplicaciones web. Entre sus características fundamentales están: separar la lógica del negocio, la lógica de servidor y la presentación de la aplicación web. Proporciona varias herramientas y clases encaminadas a reducir el tiempo de desarrollo de una aplicación web compleja. Además, automatiza las tareas más comunes, por permitir al desarrollador dedicarse por completo a los aspectos específicos de cada aplicación. Debido a estas ventajas no se reinventa una nueva aplicación web. (24)

Symfony está desarrollado en PHP 5, brindando una serie de librerías y clases para la programación web, es open source, se ejecuta en los sistemas Linux, Unix y Microsoft Windows, además de ser compatible con muchos gestores de bases de datos.

Symfony se diseñó para que se ajustara a los siguientes requisitos:

- 
- Fácil de instalar y configurar en la mayoría de plataformas (y con la garantía de que funciona correctamente en los sistemas Windows y Unix).
  - Independiente del sistema gestor de bases de datos.
  - Sencillo de usar en la mayoría de casos, pero lo suficientemente flexible como para adaptarse a los casos más complejos.
  - Basado en la premisa de "convenir en vez de configurar", en la que el desarrollador solo debe configurar aquello que no es convencional.
  - Sigue la mayoría de mejores prácticas y patrones de diseño para la web.
  - Preparado para ser utilizado en aplicaciones empresariales así como adaptable a las políticas y arquitecturas propias de cada empresa, además de ser lo suficientemente estable como para desarrollar aplicaciones a largo plazo.
  - Código fácil de leer que incluye comentarios de PHP Documentor y que permite un mantenimiento muy sencillo.
  - Fácil extensión, lo que permite su integración con librerías desarrolladas por terceros. (25)

Otro de los framework que destaca es **Zend** framework patrocinado por **Zend Technologies** para el desarrollo de aplicaciones web. Es ejecutado en versiones de PHP 5 en adelante, utiliza el patrón arquitectónico Modelo Vista Controlador (MVC), es software libre, y completamente orientado a objetos.

Aunque se pueden utilizar de forma individual, los componentes de la biblioteca estándar de Zend conforman un potente y extensible framework de aplicaciones web al combinarse. Zend Framework ofrece un buen rendimiento, una abstracción de base de datos fácil de usar, y un componente de formularios que implementa la prestación de formularios HTML, validación y filtrado para que los desarrolladores puedan consolidar las operaciones usando de una manera sencilla la interfaz orientada a objetos. Otros componentes, como Zend\_Auth y Zend\_Acl, proveen autenticación de usuarios y autorización diferentes a las tiendas de certificados comunes. También existen componentes que implementan bibliotecas de cliente para acceder de forma sencilla a los servicios web más populares. Cualesquiera que sean las necesidades de desarrollo es posible encontrar un componente de Zend Framework que se puede utilizar para reducir drásticamente el tiempo de desarrollo, con una base completamente sólida. (26)

---

### 1.3.2 Entorno de Desarrollo Integrado

Un Entorno de Desarrollo Integrado (IDE, por sus siglas en inglés) es un programa informático que da soporte a uno o varios lenguajes de programación que se van a utilizar en un determinado proyecto. Un IDE brinda un entorno adecuado para la correcta estructuración del código, brindando editores, administradores de archivos y depuradores. Existen hoy en el mercado una gran variedad de entornos de este tipo, algunos de ellos orientados específicamente a las necesidades de los programadores. Dentro de los IDEs destacados actualmente se pueden mencionar NetBeans y Eclipse, los dos herramientas de software libre.

Uno de los IDEs más conocidos hoy es **NetBeans IDE**, este surge en el año 2000 como un proyecto de software libre de la empresa Sun Micro System, ha estado en constante desarrollo y todavía sigue lanzando versiones y mejoras pues cuenta con una comunidad creciente en el mundo. Es un entorno altamente orientado a los usuarios de forma intuitiva, que puede ser utilizado muy fácilmente para el uso de proyectos web y de todas sus herramientas. Permite a los programadores escribir, compilar, depurar y ejecutar programas. Está escrito en Java pero puede servir para cualquier otro lenguaje de programación. Existe además un número importante de módulos para extenderlo.

NetBeans ofrece soporte para algunos frameworks y varias características que pueden ser de gran utilidad a la hora de desarrollar aplicaciones, entre las cuales se pueden resaltar:

- Framework para la creación de interfaces de usuario.
- Editor de datos.
- Interfaz de usuario para la personalización de la aplicación.
- Sistema de datos que permite obtener información de diferentes orígenes de datos.
- Soporte a la internacionalización de ayudas contextuales del sistema. (26)

**Eclipse** es una plataforma de desarrollo abierta compuesta por marcos extensibles, herramientas y tiempos de ejecución para la construcción, implementación y administración de software a través de todo el ciclo de vida. El Proyecto Eclipse fue creado originalmente por IBM en noviembre del año 2001 y apoyado por un consorcio de proveedores de software. En enero de 2004 se creó la Fundación Eclipse, como una organización independiente sin fines de lucro para actuar como administrador de la

---

comunidad Eclipse. Un aspecto importante de Eclipse es el objetivo de facilitar el uso de la tecnología de código abierto de los productos de software y servicios comerciales, bajo la Licencia Pública Eclipse.

A pesar del gran número de características estándar, Eclipse se diferencia de los IDEs tradicionales como Zend Studio. Lo más interesante de Eclipse es que es completamente neutral a la plataforma y al lenguaje, además de la mezcla ecléctica de lenguajes soportados por el Eclipse Consortium (Java, C/C++, Cobol), también hay otros proyectos para añadir a Eclipse el soporte de lenguajes tan diversos como Python, Eiffel, PHP, Ruby, y C#. Con respecto a las plataformas, el Consorcio Eclipse proporciona instalaciones para Windows, Linux, Solaris, HP-UX, AIX, QNX, y Mac OS X. La mayoría de los elementos que posee Eclipse se centran en la arquitectura de plug-ins y el API proporcionado por el Plug-in Development Environment para ampliarlo. Añadir soporte para un nuevo tipo de editor, una vista, o un lenguaje de programación, es fácil, con el API y los bloques de construcción que proporciona Eclipse. (27)

### 1.3.3 Sistemas de Gestión de Base de Datos

Los **Sistemas de Gestión de Base de Datos** (SGDB, por sus siglas en inglés) son herramientas informáticas para la organización estructurada de datos. Las características, conceptos y tecnologías de estos sistemas son variados. Se analizarán en este trabajo aquellas SGDB de software libre ya que permiten pautas de seguridad en el sistema operativo (no virus), desarrollo máximo para plataformas bases, flexibilidad antes los cambios. Entre los SGBD de más amplio uso se pueden mencionar PostgreSQL, MySQL y Firebird.

**MySQL** es un SGDB de modelo relacional, multihilo y multiusuario creado por la empresa Sun Microsystem. Es un gestor multiplataforma y está escrito en los lenguajes C y C++. Es de fácil uso y alto rendimiento, por lo que es atractivo y accesible a cualquier usuario; pero para mantener este alto rendimiento es necesario que el desarrollo se haga más lento.

MySQL es desarrollado como software libre bajo un licenciamiento dual. Por un lado se ofrece bajo la GNU GPL para cualquier uso compatible con esta licencia, pero para aquellas empresas que quieran incorporarlo en productos privativos deben comprar a la empresa una licencia específica que les permita este uso. Está desarrollado en su mayor parte en ANSI C. (28)

---

Entre sus características distintivas están:

- Múltiples motores de almacenamiento (MyISAM, Merge, InnoDB, BDB, Memory/heap, MySQL Cluster, Federated, Archive, CSV, Blackhole y Example en 5.x), permitiendo al usuario escoger el que sea más adecuada para cada tabla de la base de datos.
- Agrupación de transacciones que reúnen múltiples transacciones de varias conexiones para incrementar el número de transacciones por segundo. (29)

**PostgreSQL** es un gestor de base de datos de código abierto bajo la licencia Berkeley Software Distribution (BSD, por sus siglas en inglés), es el sistema de gestión de base de datos más avanzado dentro de los sistemas de software libre. Desde sus inicios en 1982 en la Universidad de Berkeley, en California, ha lanzado muchas versiones, actualmente siguen saliendo anualmente lanzamientos importantes y otros de menos relevancia pero igualmente trascendentes. Es ejecutable en muchos sistemas operativos como Linux, Unix, BSDs, Mac OS, Beos y Windows y soporte en los lenguajes PHP, C, C++, Perl, Python. Entre sus principales características están:

- Atomicidad (Indivisible): es la propiedad que asegura que la operación se ha realizado o no, y por lo tanto ante un fallo del sistema no puede quedar a medias.
- Consistencia: es la propiedad que asegura que sólo se empieza aquello que se puede acabar. Por lo tanto se ejecutan aquellas operaciones que no van a romper la reglas y directrices de integridad de la base de datos.
- Aislamiento: es la propiedad que asegura que una operación no puede afectar a otras. Esto asegura que dos transacciones sobre la misma información nunca generarán ningún tipo de error.
- Durabilidad: es la propiedad que asegura que una vez realizada la operación, esta persistirá y no se podrá deshacer aunque falle el sistema. (30)

**Firebird** es un SGDB relacional, multiplataforma, de alto rendimiento, amplios procedimientos almacenados y disparadores. Es creado con código de Interbase 6.0 de Borland también liberado bajo la Licencia Pública Interbase.

Firebird tiene todas las características y la potencia de un RDBMS. Se pueden manejar bases de datos desde unos pocos kb hasta varios gigabytes con buen rendimiento y casi sin mantenimiento. (31)

Entre sus principales características están:

- 
- Multiplataforma.
  - Ejecutables pequeños con requerimientos de hardware mínimos.
  - Arquitectura Cliente/Servidor bajo el protocolo TCP.
  - Integridad referencial.
  - Soporte para funciones externas.
  - Fácil de instalar y usar.

#### 1.3.4 Servidor Web

Un **Servidor Web** puede definirse como un programa que escucha las peticiones de los usuarios o navegantes y las atiende o satisface. Por medio de la especificación de la búsqueda el servidor Web busca una página específica o ejecutará un programa (32). Entre los servidores de más amplio uso se pueden mencionar IIS y Apache.

**Apache Web Server** es un servidor web de software libre que se destaca por su robustez y a la vez estabilidad, presenta muchas opciones de ayuda a los usuarios que lo hacen muy configurable. Presenta un diseño modular lo que le permite a los usuarios insertarle nuevas funcionalidades o ampliar las ya existentes para adaptarse más a las necesidades de cada institución en particular. Es multiplataforma y trabaja con lenguajes como Perl, PHP y Java. Permite conexiones seguras bajo el protocolo Hypertext Transitional Protocol Safety (https, por sus siglas en inglés), además de corrección de errores y posibles respuestas ante fallos. Por todas sus características y adaptabilidad posee amplia aceptación mundial.

**Servidor Web Internet Information Server** (IIS, por sus siglas en inglés) es el servidor web por defecto del sistema operativo Microsoft Windows y de transferencia de archivos segura bajo el protocolo https. Posee arquitectura modular que le posibilitan procesar distintos tipos de páginas como Active Server Pages (ASP) y ASP.NET. Su equipo de desarrollo ha trabajado en mejoras de seguridad y rendimiento, pero sigue limitado por plataformas Windows y su carácter de software propietario. Está destinado a publicar y gestionar páginas web.

Los servicios IIS, son los servicios de software que admiten la creación, configuración y administración de sitios Web, además de otras funciones de Internet. Los servicios de Microsoft Internet Information Server incluyen el Protocolo de transferencia de noticias a través de la red (NNTP, por sus siglas en inglés), el Protocolo de transferencia de

---

archivos (FTP, por sus siglas en inglés) y el Protocolo simple de transferencia de correo (SMTP, por sus siglas en inglés). (33)

**XAMPP** es un servidor independiente de plataforma y trabaja con lenguajes como Perl y PHP. De igual forma trabaja con la base de datos MySQL y el servidor web Apache. Está bajo la licencia GNU y actúa como un servidor web libre, es fácil de usar y capaz de interpretar páginas dinámicas.

### 1.3.5 Metodologías de desarrollo de software

Las **metodologías de desarrollo de software** son un conjunto de procedimientos, técnicas y ayudas a la documentación para el desarrollo de productos software. (34)

Existen varias clasificaciones de metodologías, pero no se puede afirmar que con una metodología específica se pueda guiar satisfactoriamente un proyecto completo, sino que estas se deben adaptar a necesidades del proyecto como son tiempo, recursos, equipo de desarrollo y tipo de sistema.

En ingeniería del software aproximadamente cada década aparece un nuevo tipo de metodologías. En los 70 del siglo pasado, fue el desarrollo estructurado frente al tradicional. En los 80, el desarrollo dirigido por datos frente al tradicional. En los 90, el desarrollo orientado a objetos frente al tradicional y al dirigido por datos. Y a principios de los 2000, el ágil frente al desarrollo tradicional y el CMMI. (35)

Actualmente prevalecen las clasificaciones de metodologías ágiles y tradicionales. Las metodologías tradicionales o pesadas se caracterizan por estar basadas en normas de estándares seguidas por el entorno de desarrollo, controladas por muchas normas y políticas, creación de gran número de artefactos y roles por lo que precisa de un equipo de desarrollo grande donde el cliente interactúa mediante reuniones, es resistente ante cambios y otorga gran importancia a la arquitectura del software, expresándola a través de modelos. Mientras que las ágiles están especialmente preparadas para requisitos cambiantes, procesos menos controlados con menos roles y artefactos, donde el cliente es parte del equipo de desarrollo y se le hace menos énfasis a la arquitectura del software.

Dentro de las **Metodologías de Desarrollo de Software Tradicionales** la más establecida es **Rational Unified Process** (RUP, por sus siglas en inglés), esta es una metodología para el desarrollo de un proyecto de un software que define claramente

---

quién, cómo, cuándo y qué debe hacerse en el proyecto. Como características esenciales se pueden decir que RUP es dirigido por Casos de Uso: que orientan el proyecto a la importancia para el usuario y lo que este quiere; está centrado en la arquitectura: relaciona la toma de decisiones que indican cómo tiene que ser construido el sistema y en qué orden, además es iterativo e incremental: divide el proyecto en mini proyectos donde los casos de uso y la arquitectura cumplen sus objetivos de manera más depurada. (36)

RUP divide en cuatro fases el desarrollo de software con sus respectivos hitos, estas fases son: Inicio, Elaboración, Construcción y Transición. En cada una de estas fases se crean artefactos de software para comprender el análisis y diseño del sistema. Brinda una guía fundamental para la realización y asignación de tareas responsablemente, abarca las mejores prácticas de Ingeniería de Software, como son control de cambio, modelado visual del software, verificación de la calidad del mismo y creación de artefactos claves en el entendimiento del negocio del sistema a través del uso del Lenguaje Unificado de Modelado. Esta metodología propone contar con un grupo grande de desarrollo para trabajar, ya que hace análisis exhaustivos en cada fase y genera gran cantidad de documentación.

Dentro de las **Metodologías Ágiles** se encuentran XP, Crystal y AUP.

**Extreme Programming** (XP, por sus siglas en inglés) está basada en la adaptabilidad, potencia las relaciones entre el equipo de desarrollo y la relación entre cliente-equipo de desarrollo como comunicación fundamental para el flujo de ideas y tareas colaborativas que contribuyan al éxito. XP está orientada a aquellos sistemas con requisitos volátiles y cuenta con 6 fases: Exploración, Planificación, Iteraciones, Producción, Mantenimiento y Muerte del Proyecto. Entre sus características principales se encuentran:

- El desarrollo iterativo e incremental: pequeñas mejoras unas tras otras, realización de pruebas unitarias continuas, frecuentemente repetidas y automatizadas, incluyendo pruebas de regresión; programación en parejas.
- La frecuente integración del equipo de programación con el cliente o usuario,
- La corrección de todos los errores antes de añadir nueva funcionalidad, realizar entregas frecuentes.
- La simplicidad en el código: es la mejor manera de que las cosas funcionen. Cuando todo funcione se podrá añadir funcionalidad si es necesario. La programación extrema apuesta que es más sencillo hacer algo simple y tener un

---

poco de trabajo extra para cambiarlo si se requiere, que realizar algo complicado y quizás nunca utilizarlo. La simplicidad y la comunicación son extraordinariamente complementarias. Con más comunicación resulta más fácil identificar qué se debe y qué no se debe hacer. Mientras más simple es el sistema, menos tendrá que comunicar sobre este, lo que lleva a una comunicación más completa, especialmente si se puede reducir el equipo de programadores. (36)

**Crystal** es un conjunto de metodologías que se centran en las personas que conforman el equipo de desarrollo y la reducción de artefactos, destacando la cooperación y actividades de capacitación en sus integrantes.

Los métodos se llaman Crystal evocando las facetas de una gema: cada faceta es otra versión del proceso, y todas se sitúan en torno a un núcleo idéntico. Hay cuatro variantes de metodologías: Crystal Clear para equipos de 8 o menos integrantes; Amarillo, para 8 a 20; Naranja, para 20 a 50; Rojo para 50 a 100. (37)

Crystal tiene como características:

- Combinar la productividad y la tolerancia.
- Trabajar con equipos pequeños, logrando mejor comunicación entre los equipos del proyecto.
- Dar vital importancia a las personas que componen su equipo de trabajo, teniendo como puntos de estudio:
  - Aspecto humano del equipo.
  - Tamaño de un proyecto.
  - Comunicación dentro del equipo.
  - Distintas políticas a seguir.
  - Espacio físico de trabajo. (38)

La metodología **Proceso Ágil Unificado**, más conocida como AUP, se define como una versión simplificada de la metodología pesada RUP. Esta resume de la forma más simple posible las formas de desarrollar aplicaciones de manera sencilla haciendo uso de las metodologías ágiles, y sin perder conceptos que se mantienen en RUP.

Al igual que en RUP se trazan cuatro fases fundamentales que guíen el proceso de desarrollo: Inicio, Elaboración, Construcción y Transición, que son iterativas y culminan con la entrega de un hito definido, ofrece especial atención en la gestión de riesgos.

La AUP es ágil, porque está basada en los siguientes principios:

- 
1. El personal sabe lo que está haciendo. La gente no va a leer detallado el proceso de documentación, pero algunos quieren una orientación de alto nivel y / o formación de vez en cuando. La AUP proporciona enlaces a muchos de los detalles, si usted está interesado, pero no obliga a aquellos que no lo deseen.
  2. Simplicidad. Todo se describe concisamente utilizando un puñado de páginas, no miles de ellas.
  3. Agilidad. Ágil arriba el ajuste a los valores y principios de la Alianza Ágil.
  4. Centrarse en actividades de alto valor. Herramienta de la independencia. Usted puede usar cualquier conjunto de herramientas que usted desea con el AUP. Lo aconsejable es utilizar las herramientas que sean más adecuadas para el trabajo, que a menudo son las herramientas simples o incluso herramientas de código abierto.
  5. Adaptación de este producto para satisfacer sus propias necesidades. El producto AUP es de fácil acomodo común a través de cualquier herramienta de edición de HTML. No se necesita comprar una herramienta especial, o tomar un curso, para adaptar la AUP. (39)

#### **1.4 Conclusiones parciales**

Luego de un estudio de los principales elementos vinculados al objeto y al campo de investigación, producto de una revisión bibliográfica se arriban a las siguientes conclusiones:

1. La autogestión del conocimiento, como capacidad de los estudiantes de autorregular su aprendizaje es una habilidad importante a alcanzar como contribución a la formación de profesionales integrales en la UCI.
2. Los sistemas de gestión del conocimiento de forma general deben forjar procesos colaborativos, distribuir y sincronizar tareas en la organización, de forma tal que se pueda reducir el tiempo y aumentar la eficacia.
3. En los CASIE no existe un sistema que le permita al profesor establecerle un tratamiento diferenciado a cada estudiante, creando las bases para la autogestión del conocimiento desde el idioma inglés.
4. Existen un conjunto de herramientas, tecnologías y metodologías que guían el proceso de desarrollo de aplicaciones web, su estudio, en el capítulo que concluye, permitirá la selección posterior de las adecuadas para construir la aplicación requerida.

---

## Capítulo II: Exploración y Planificación.

Este capítulo tiene por objetivos definir las características generales del sistema, los procesos a automatizar y realizar la descripción de los requisitos; además de lo anterior se deben seleccionar los lenguajes, herramientas, metodologías y otros elementos que permitirán llevar a cabo la propuesta, realizando la planificación del mismo.

En la UCI se hace necesario el dominio del idioma Inglés por sus estudiantes, debido a que este es un requisito para formar un profesional integral preparado para ejercer su profesión en cualquier país del mundo, además los estudiantes en la universidad se ven en constante búsqueda de bibliografías actualizadas necesarias en sus asignaturas las cuales se encuentran en su mayoría en idioma Inglés, así como sus búsquedas en Internet donde un alto por ciento de los archivos están en este idioma.

La UCI brinda a sus estudiantes diferentes recursos para el estudio del inglés, como ejemplo se pueden mencionar los CASIE. Estos centros pueden ser visitado por los estudiantes en cualquier horario del día, acceder a sus materiales, y en caso de estar presente un asesor preguntar dudas. Los asesores no pueden estar todo el tiempo en los CASIE por lo que la mayoría de las veces el alumno se encuentra solo y no aprovecha debidamente su estancia, pues estos centros no cuentan con un sistema específico para motivar y proporcionarles adecuadamente los conocimientos a la comunidad universitaria que visita el lugar. Sería de gran utilidad el desarrollo de un sistema que permita a los asesores desenvolver las habilidades de gestión del conocimiento en los estudiantes sin necesidad de varios encuentros y garantizando además el seguimiento individual de cada estudiante, orientándole las tareas según sus avances en el idioma. Como vía de solución al problema planteado anteriormente es que se propone realizar una primera versión de un Sistema de Apoyo para la Autogestión del Conocimiento en los Centros de Autoaprendizaje y Servicios de Idiomas Extranjeros (LanGlish), el mismo organizará de forma adecuada los indicadores que permiten crear un sistema para la gestión del conocimiento, de forma interactiva y amena, para que los usuarios que visitan el sistema puedan no solo estudiar sino lograr la autogestión del conocimiento.

### 2.1 Objeto de automatización

El estudio de la comunidad universitaria que visita los CASIE actualmente se realiza con la orientación del trabajador del CASIE el cual le orienta materiales generales a estudiarse, teniendo que consultar constantemente al técnico del centro en casos de

---

dudas o nuevas orientaciones, además de que los profesores de lenguas extranjeras no se encuentran en el CASIE la mayoría de las veces. Con la implementación de LanGlish los estudiantes podrán interactuar con una plataforma web sin la orientación constante de técnicos ni profesores, pues este sistema le da tratamiento diferenciado a los estudiantes según su año y nivel de autogestión. Se adapta a las necesidades específicas del usuario y no les hace divagar en contenidos que domine de antemano. El sistema está concebido para que el estudio del contenido sea más ameno, pues ofrecerá interactividad, fácil navegabilidad y opciones de gestión de usuarios útiles y agradables.

## **2.2 Descripción del sistema**

Para dar solución al problema planteado se desarrollará un sistema, que dependiendo del año que curse el estudiante le será realizado un diagnóstico donde cuyo resultado permitirá saber el nivel de autogestión del conocimiento que tiene. El diagnóstico constará de tres fases utilizando como base las habilidades que según Prieto (10) propician la autogestión de la información, es decir, la búsqueda, selección y filtrado de la información. El estudiante será ubicado según su año y nivel de autogestión en un grupo determinado donde se le guiará para que cumpla con los objetivos y adquiera las habilidades que le faltan para poder autogestionar el conocimiento mediante el idioma inglés vinculando las diferentes disciplinas de la carrera. Luego que el estudiante se encuentre en el grupo asignado se trabajará además en la comprensión y aplicación de la información mediante tareas y talleres que le serán propuestos.

En todo momento el usuario cuenta con la posibilidad de autoevaluarse y autorregular su actividad, consultando al asesor y redefiniendo su tratamiento para el aprendizaje y estudio de la disciplina. Siendo esta la primera versión del sistema que le será entregado a los CASIE para iniciar el proceso de autogestión del conocimiento desde estos centros.

### **2.2.1 Modelo de dominio**

El modelo de dominio o conceptual puede utilizarse para capturar y expresar el entendimiento ganado en un área bajo análisis como paso previo al diseño de un sistema. El modelo de dominio es utilizado por el analista como un medio para comprender el sector de negocios al cual el sistema va a servir, para llegar a un entendimiento común que permita crear las bases para el desarrollo.

---

## 2.2.2 Realización del modelo de dominio

Se identificaron los conceptos y clases que se muestran en el modelo de dominio a continuación:

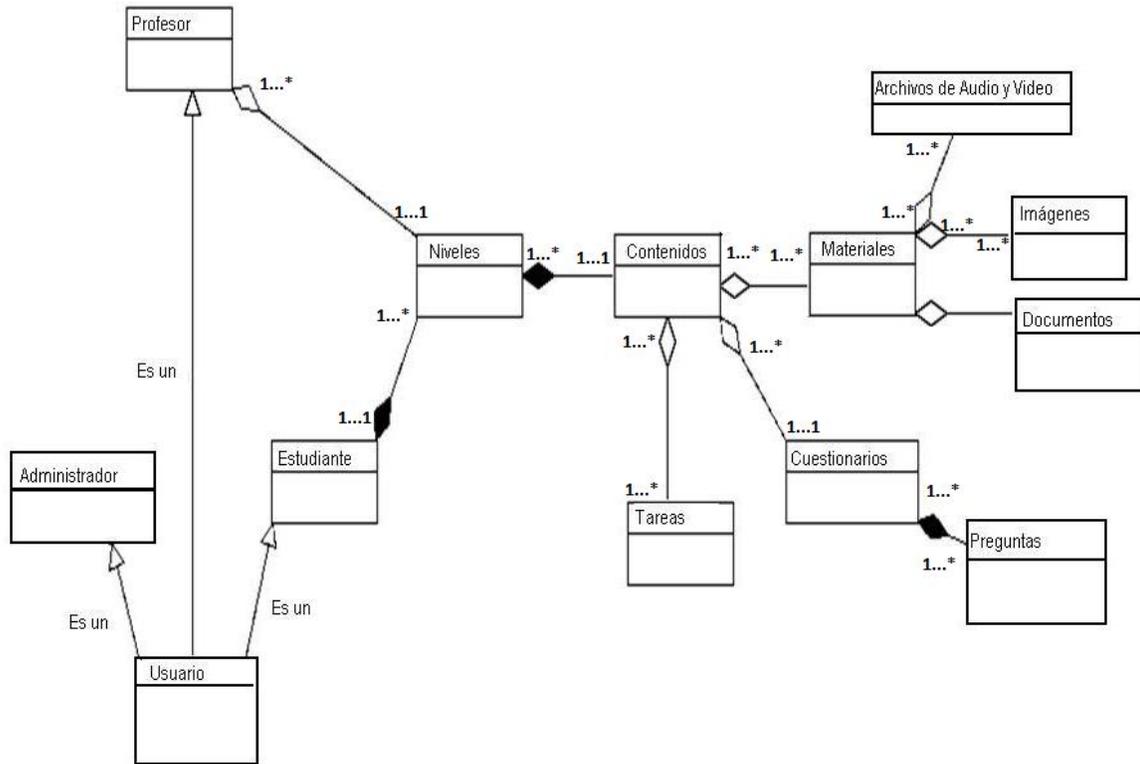


Figura 1: Modelo de Dominio

## 2.3 Exploración del Sistema

### 2.3.1 Historias de usuarios

Las Historias de Usuario (HU) es uno de los artefactos generados por la metodología XP, los mismos ofrecen una breve descripción del comportamiento del sistema. Estas HU proporcionan una información muy básica, solo la necesaria para realizar estimaciones de utilidad a la hora de la implementación. Se puede afirmar que las HU son artefactos no formales, ya que las construyen los clientes empleando su propia terminología sin conocimientos esenciales sobre el desarrollo, es decir sin lenguaje técnico. Son fundamentales para realizar estimaciones de tiempo, además de guiar el proceso de creación de las pruebas de aceptación.

Las HU se modelan a través de tablas que contienen los siguientes campos:

- Número: número de la HU incremental en el tiempo.
- Nombre de Historia de Usuario: el nombre de la HU para lograr una mejor identificación.
- Modificación de Historia de Usuario Número: historial que refleje si ha tenido modificaciones anteriores.
- Usuario: nombra a aquellas personas involucradas en el desarrollo de la HU.
- Iteración Asignada: número de la iteración.
- Prioridad en negocio: Muy Alta, Alta, Media o Baja, según la prioridad de sus funcionalidades.
- Riesgo en Desarrollo: Muy Alta, Alta, Media o Baja, según la existencia de posibles errores y su impacto en el sistema.
- Puntos estimados: tiempo estimado para el desarrollo de la HU.
- Puntos Reales: tiempo real usado para el desarrollo de la HU.
- Descripción: breve descripción de la HU.
- Observaciones: observaciones sobre la HU en el sistema.
- Prototipo de interfaz: prototipo de interfaz de usuario si aplica.

A continuación se muestran las tablas correspondientes a aquellas HU del sistema clasificadas de acuerdo a su prioridad en el negocio y su riesgo de desarrollo en muy altas. Las demás pueden ser consultadas en el Anexo 3 de este documento.

Historia de Usuario	
<b>Código:</b> HU-3	<b>Nombre de Historia de Usuario:</b> Gestionar Preguntas.
<b>Modificación de Historia de Usuario Número:</b> Ninguna	
<b>Usuario:</b> Dariela Ferreiro Elayne Elejalde	<b>Iteración Asignada:</b> 1
<b>Prioridad en negocio:</b> Alta	<b>Puntos estimados:</b> 1
<b>Riesgo en Desarrollo:</b> Alto	<b>Puntos Reales:</b> 1
<b>Descripción:</b> El editor entra al sistema y selecciona la opción Gestionar Preguntas, el sistema permite crear/modificar/eliminar/ver preguntas.	
<b>Observaciones:</b> El editor debe estar autenticado y con los permisos para gestionar las preguntas.	

**Tabla 1: HU #1: Gestionar Preguntas**

Historia de Usuario	
<b>Código:</b> HU-4	<b>Nombre de Historia de Usuario:</b> Gestionar Cuestionarios.
<b>Modificación de Historia de Usuario Número:</b> Ninguna	
<b>Usuario:</b> Dariela Ferreiro Elayne Elejalde	<b>Iteración Asignada:</b> 1
<b>Prioridad en negocio:</b> Alta	<b>Puntos estimados:</b> 1
<b>Riesgo en Desarrollo:</b> Alto	<b>Puntos Reales:</b> 1
<b>Descripción:</b> El editor entra al sistema y selecciona la opción Gestionar Cuestionarios, el sistema	

permite crear/modificar/eliminar/ver cuestionarios.
<b>Observaciones:</b> El editor debe estar autenticado y con los permisos para gestionar los cuestionarios.

**Tabla 2: HU #4: Gestionar Cuestionarios**

Historia de Usuario	
<b>Código:</b> HU-5	<b>Nombre de Historia de Usuario:</b> Gestionar Diagnósticos.
<b>Modificación de Historia de Usuario Número:</b> Ninguna	
<b>Usuario:</b> Dariela Ferreiro Elayne Elejalde	<b>Iteración Asignada:</b> 1
<b>Prioridad en negocio:</b> Alta	<b>Puntos estimados:</b> 1
<b>Riesgo en Desarrollo:</b> Alto	<b>Puntos Reales:</b> 1
<b>Descripción:</b> El editor entra al sistema y selecciona la opción Gestionar Diagnósticos, el sistema permite crear/modificar/eliminar/ver diagnósticos.	
<b>Observaciones:</b> El editor debe estar autenticado y con los permisos para gestionar los diagnósticos.	

**Tabla 3: HU #5: Gestionar Diagnósticos**

Historia de Usuario	
<b>Código:</b> HU-7	<b>Nombre de Historia de Usuario:</b> Gestionar Materiales.
<b>Modificación de Historia de Usuario Número:</b> Ninguna	
<b>Usuario:</b> Dariela Ferreiro Elayne Elejalde	<b>Iteración Asignada:</b> 1
<b>Prioridad en negocio:</b> Alta	<b>Puntos estimados:</b> 1
<b>Riesgo en Desarrollo:</b> Alto	<b>Puntos Reales:</b> 1
<b>Descripción:</b> El editor entra al sistema y selecciona la opción Gestionar Materiales, el sistema permite adjuntar/modificar/eliminar/ver materiales.	
<b>Observaciones:</b> El editor debe estar autenticado y con los permisos para gestionar los materiales.	

**Tabla 4: HU #7: Gestionar Materiales**

Historia de Usuario	
<b>Código:</b> HU-11	<b>Nombre de Historia de Usuario:</b> Asignar Niveles.
<b>Modificación de Historia de Usuario Número:</b> Ninguna	
<b>Usuario:</b> Dariela Ferreiro Elayne Elejalde	<b>Iteración Asignada:</b> 1
<b>Prioridad en negocio:</b> Alta	<b>Puntos estimados:</b> 1
<b>Riesgo en Desarrollo:</b> Alto	<b>Puntos Reales:</b> 1
<b>Descripción:</b> El estudiante realiza un cuestionario de nivel y luego lo envía. El sistema realiza la asignación de nivel.	
<b>Observaciones:</b> Se debe enviar el cuestionario luego de ser resuelto.	

**Tabla 5: HU #11: Asignar Niveles**

### 2.3.2 Requisitos funcionales

Los requisitos funcionales (condiciones o capacidades que debe cumplir el producto) son especificados por el cliente y agrupados según su prioridad en las categorías: muy alta, alta, media o baja, con prioridad descendente en el orden especificado, de acuerdo a la necesidad de su realización para el funcionamiento del sistema.

Código	Requisitos Funcionales	Prioridad
HU-1	Gestionar Roles	
1	Crear Rol	Alta
2	Modificar Rol	
3	Eliminar Rol	
4	Asignar Rol	
5	Ver Rol	
HU-2	Gestionar Tópicos	
6	Crear Tópicos	Alta
7	Ver Tópicos	
8	Modificar Tópicos	
9	Eliminar Tópicos	
HU-3	Gestionar Preguntas	
10	Crear Preguntas	Alta
11	Modificar Preguntas	
12	Eliminar Preguntas	
13	Ver Preguntas	
HU-4	Gestionar Cuestionarios	
14	Crear Cuestionarios	Alta
15	Modificar Cuestionarios	
16	Eliminar Cuestionarios	
17	Ver Cuestionarios	
HU-5	Gestionar Diagnóstico	
18	Crear Diagnóstico	Alta
19	Modificar Diagnóstico	
20	Ver Diagnóstico	
21	Eliminar Diagnóstico	
HU-6	Realizar cuestionarios	
22	Guardar sin enviar	Alta
23	Evaluar Cuestionarios	
HU-7	Gestionar Materiales	
24	Adjuntar Materiales	Alta
25	Modificar Materiales	
26	Eliminar Materiales	
27	Ver Materiales	
HU-8	Gestionar Tareas	

28	Crear Tarea	Alta
29	Modificar Tareas	
30	Eliminar Tareas	
31	Ver Tareas	
HU-9	Revisar Tareas Asignadas	Alta
HU-10	Realizar Tareas	Alta
HU-11	Asignar Niveles	Muy Alta
HU-12	Autenticar Usuario	Alta
HU-13	Notificación de actividades	Media
HU-14	Ver Perfiles de Usuarios	Alta
HU-15	Gestionar Grupos	
32	Crear Grupos	Alta
33	Ver Grupos	
34	Modificar Grupos	
35	Eliminar Grupos	
HU-16	Gestionar Orientaciones de Trabajo	
36	Crear Orientaciones de Trabajo	Alta
37	Ver Orientaciones de Trabajo	
38	Modificar Orientaciones de Trabajo	
39	Eliminar Orientaciones de Trabajo	
HU-17	Gestionar Contenidos	
40	Crear Contenidos	Alta
41	Ver Contenidos	
42	Modificar Contenidos	
43	Eliminar Contenidos	

**Tabla 6: Requisitos Funcionales**

### 2.3.3 Requisitos no funcionales

Los requisitos no funcionales son las cualidades o propiedades del software para alcanzar una buena aceptación.

<b>Requisitos No Funcionales</b>	
Hardware	Las máquinas clientes necesitan de una estación de trabajo con procesador Pentium II en adelante, 128 megabytes (MB) de memoria RAM mínimo y poseer conexión de red a velocidad de 100 megabytes por segundo (Mbps). La máquina servidor necesita un procesador Pentium II en adelante, 256 megabytes (MB) de memoria RAM mínimo, poseer conexión de red a velocidad de 100 Mbps, capacidad libre en disco duro de 1 GB.
Software	Las máquinas clientes necesitan sistema operativo Windows o Linux en cualquiera de sus distribuciones, un navegador que soporte el estándar W3C (Consortio World Wide Web) como Mozilla Firefox, Internet Explorer 7 en adelante, Opera, Safari o Google Chrome. La máquina servidor necesita sistema operativo Windows o Linux en

	cualquiera de sus distribuciones, servidor Apache, SGDB PostgreSQL, framework Symfony.
Apariencia o interfaz externa.	Debe presentar una interfaz intuitiva, accesible, con un diseño atractivo al usuario, con dimensiones mínimas de 1024x768 píxeles para adaptarse a diferentes resoluciones de pantalla.
Usabilidad	Será diseñado con una correcta estructuración de la información contenida en la misma para una fácil navegabilidad en el menor tiempo posible, contando con mapas de navegación, iconos que puedan servir de pistas visuales y ayuda.
Rendimiento	Debe mantener un tiempo de respuesta rápido ante cualquier acción, soportando gran cantidad de usuarios conectados a la vez interactuando con el mismo.
Soporte	Será probado previamente e instalado por el equipo de desarrollo, además preparar personal capacitado para administrar el sitio, configurarlo y ofrecerle mantenimiento de acuerdo a las peticiones del cliente.
Portabilidad	Se debe ejecutar sobre sistemas operativos Windows y Linux.
Seguridad	Para asegurar la confidencialidad solo tendrán acceso a información confidencial el personal autorizado, la misma será almacenada en bases de datos con registros seguros, se realizarán trazas y filtros para detectar problemas y mantener la integridad de los datos. Se incluirán en algún apartado del sitio las políticas de uso y seguridad para que puedan ser consultadas por todos los usuarios así como lograr que estén disponibles junto con toda la información las 24 horas del día.
Legales	El producto de software así como los artefactos generados son propiedad de la UCI.
Confiabilidad	Debe dar respuesta a los usuarios con la mayor certeza posible en el mínimo tiempo.
Ayuda y documentación en línea	Debe tener disponible en todo momento documentos de ayuda y navegación a disposición de los usuarios que puedan presentar dudas.

**Tabla 7: Requisitos No Funcionales**

## 2.4 Selección de metodologías, tecnologías y herramientas

Luego de haber descrito las características del sistema, se han seleccionado para el desarrollo del mismo:

### 2.4.1 Metodología de Desarrollo de Software: XP

Se seleccionó la metodología XP para conducir todos los procesos y actividades que guíen el software durante todo su ciclo de vida. La misma es una metodología ágil orientada a equipos de desarrollo pequeños, con plazos de tiempo reducidos para el desarrollo, requisitos cambiantes, y apoya el uso de tecnologías innovadoras.

---

## **2.4.2 Lenguajes de Programación: PHP 5 y JavaScript 1.1**

PHP 5 se escogió como lenguaje del lado del servidor para el desarrollo de LanGlish, ya que el mismo es un lenguaje de código abierto y por tanto puede ser utilizado sin restricciones ni costos por la universidad. Como lenguaje del lado del cliente se seleccionó JavaScript 1.1, por su compatibilidad con muchos navegadores, ser multiplataforma y de alto nivel, para desarrollar todo el proceso de interacción con el usuario.

## **2.4.3 Framework de Desarrollo: Symfony 2.0**

Por sus características para el desarrollo de proyectos web se decidió escoger Symfony 2.0 para la implementación del sistema, ya que proporciona herramientas que simplificarán el desarrollo de la aplicación web. Está basado en la programación orientada a objetos, en el mapeo de objetos a bases de datos y el desarrollo rápido de aplicaciones. Permite a usuarios que no estén muy familiarizados con el framework interactuar con el mismo a través de templates en HTML, brinda enrutamiento y URL inteligentes, además de plugins extensibles y listas amigables para la paginación, clasificación y filtrado. Una característica destacable es que el framework es software libre con una amplia documentación, que cuenta con una extensa comunidad en el mundo y que está teniendo mucha difusión actualmente.

## **2.4.4 Entorno de Desarrollo Integrado: NetBeans IDE 7.1**

Se utilizará como Entorno de Desarrollo Integrado NetBeans IDE 7.1 por sus características: es una herramienta de fácil uso y muy intuitiva lo que traerá ayuda al equipo de desarrollo, proporciona autocompletado y documentación de funciones PHP, que es el lenguaje de programación que será usado para el desarrollo del sistema.

## **2.4.5 Sistema de Gestión de Base de Datos: PostgreSQL 8.4**

Para la construcción del sistema de apoyo destinado a los CASIE se decide utilizar el SGDB PostgreSQL en su versión 8.4. La mayoría de los cambios en PostgreSQL 8.4 son herramientas y órdenes de administración y monitoreo, nuevas o mejoradas. Cada usuario tiene su funcionalidad favorita que hace su trabajo cotidiano con PostgreSQL más fácil y productivo. (40)

---

#### **2.4.6 Servidor Web: Apache 2.0**

Por su gran potencialidad y aceptabilidad que ha alcanzado hoy en día el servidor Apache se seleccionó la versión 2.0 de Apache como servidor web del sistema en desarrollo. Permite adaptarse a las necesidades del software por su configurabilidad y arquitectura modular, brinda opciones para usuarios que tengan poco dominio del tema como respuestas ante errores, una buena ayuda y transferencias seguras. Además es un software gratuito y de código abierto, por lo que puede ser utilizado sin restricciones por la UCI.

#### **2.5 Descripción de la Arquitectura**

La Arquitectura de software describe la estructura y relaciones entre los elementos de un software determinado. Permite ocuparse de la organización del sistema en cuanto a su disponibilidad y control global. Según el libro "An introduction to Software Architecture" de David Garlan y Mary Shaw la Arquitectura es un nivel de diseño que realza aspectos "más allá de los algoritmos y estructuras de datos de la computación; el diseño y especificación de la estructura global del sistema es un nuevo tipo de problema".

Para el desarrollo del sistema LanGlish se adoptó la Arquitectura en Tres Niveles, o Capas como también se le conoce. Este tipo de arquitectura es ampliamente usado para la programación de aplicaciones web. Propone que la aplicación se divida en tres capas lógicas distintas, cada una de ellas con un grupo de interfaces perfectamente definidas. La primera capa se denomina capa de presentación y normalmente consiste en una interfaz gráfica de usuario de algún tipo, la capa intermedia contiene la lógica de la aplicación, y la tercera (capa de acceso a datos) es la encargada de los procesos de almacenamiento y envío de datos. Esta arquitectura permite altos niveles de abstracción donde cada nivel esconde de los demás detalles contenidos en los niveles inferiores, posibilita el desarrollo simplificado del aplicativo ya que al desarrollar un nivel no precisa preocuparse por el otro, además de mayor seguridad y protección porque el desarrollador de un nivel no puede controlar físicamente otra camada en ningún nivel de detalle.

#### **Estilo Arquitectónico**

El estilo arquitectónico que se usará es el Modelo Vista Controlador (MVC), asociado a la Arquitectura Tres Capas pero refinando específicamente el comportamiento de los componentes presentes en la misma.

---

La selección del MVC está determinada por el framework de desarrollo, Symfony 2.0 que se usará para desarrollar el sistema. Este estilo arquitectónico separa la lógica del negocio de la presentación y de los datos, haciendo más sencillo el mantenimiento de aplicaciones, mayor flexibilidad y reusabilidad.

Se definen tres componentes:

1. Modelo: administra la lógica de la aplicación y las funciones de integridad, comunicándose con la base de datos.
2. Vista: es la forma visual de gestionar el modelo al usuario.
3. Controlador: permite la interacción del usuario con el sistema.

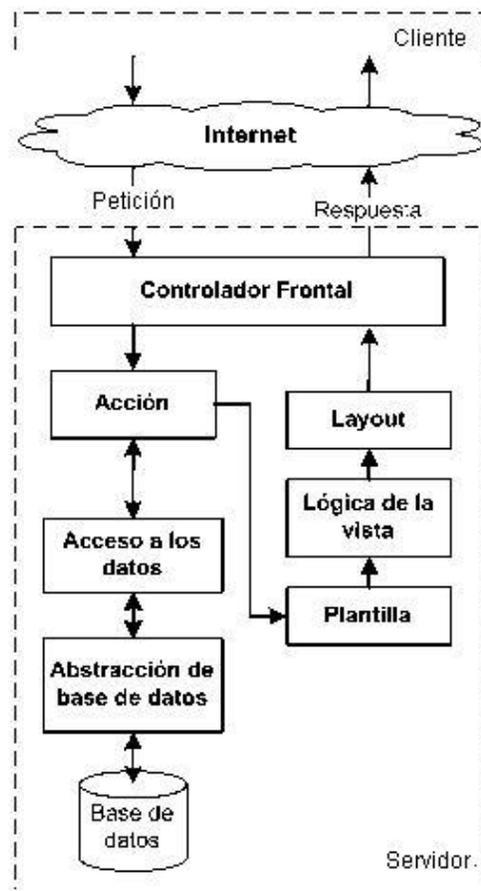
A continuación se detallan cada componente:

1. El modelo es el responsable de:
  - Acceder a la capa de almacenamiento de datos. Lo ideal es que el modelo sea independiente del sistema de almacenamiento.
  - Define las reglas de negocio (la funcionalidad del sistema).
  - Lleva un registro de las vistas y controladores del sistema.
  - Si estamos ante un modelo activo, notificará a las vistas los cambios que en los datos pueda producir un agente externo.
2. El controlador es responsable de:
  - Recibir los eventos de entrada (un clic, un cambio en un campo de texto, etc.).
  - Contiene reglas de gestión de eventos, del tipo "SI Evento Z, entonces Acción W". Estas acciones pueden suponer peticiones al modelo o a las vistas.
3. Las vistas son responsables de:
  - Recibir datos del modelo para mostrar al usuario.
  - Tienen un registro de su controlador asociado (normalmente porque además lo instancia). (41)

Symfony por su parte toma lo mejor de la arquitectura MVC y la implementa de forma que el desarrollo de aplicaciones sea rápido y sencillo. Las vistas se implementan con un controlador frontal (generado automáticamente) y plantillas comunes para toda la aplicación. Las clases del modelo son generadas también de acuerdo a la estructura de datos de la aplicación a través de la función Object Relational Mapping (ORM, por sus

---

siglas en inglés) de Symfony. La lógica se puede transformar en un archivo de configuración sencillamente.



**Figura 2: Flujo de trabajo de Symfony usando MVC**

## Patrones de Diseño

Dentro de la Arquitectura de Software, se han definido patrones de diseño para describir comportamientos para solucionar problemas particulares y recurrentes en el diseño.

Un patrón de diseño es una descripción de clases y objetos comunicándose entre sí adaptada para resolver un problema de diseño general en un contexto particular. (42)

En el mundo del software actual la utilización de patrones de diseño para desarrollar los mismos es de gran ayuda, estos permiten capturar experiencias y propician ayuda a los desarrolladores de todo el mundo, brindándoles un lenguaje común para resolver

---

problemas que se presenten durante el diseño. Además aportan reutilización y extensibilidad al software, agilizando el desarrollo y la mejor calidad del mismo.

Los patrones de diseño constan de tres elementos, el nombre: describe el problema, el problema: describe cuándo usar el patrón y la solución: describe los elementos del diseño que darán respuesta al problema, sus relaciones, responsabilidades y colaboración.

Dentro de los patrones de diseño están los Patrones para Asignar Responsabilidades y los Patrones Gang of Four (GOF, por sus siglas en inglés), muchos de los mismos son usados en el sistema.

### **Patrones para Asignar Responsabilidades**

Los patrones para Asignar Responsabilidades (GRASP, por sus siglas en inglés) describen los principios fundamentales de diseño de objetos para la asignación de responsabilidades, refiriéndose a las responsabilidades relacionadas con los objetos en cuanto a su comportamiento. Los patrones GRASP clasifican estas responsabilidades, en lo que los objetos deben "Conocer " (conocer los datos encapsulados, los objetos relacionados y las cosas que se derivan del mismo) y "Hacer " (hacer algo él mismo, iniciar, controlar o coordinar acciones en otros objetos).

Los patrones GRASP usados en el desarrollo del sistema son:

- **Experto:** es el patrón básico de asignación de responsabilidades, plantea que la creación de un objeto o la implementación de un método, debe recaer sobre la clase que conoce toda la información necesaria para crearlo. Con el uso de Symfony, y su módulo Doctrine para mapear la Base de Datos está presente este patrón, la utilización de este módulo en Symfony permite encapsular toda la lógica de los datos y generar las clases con todas las funcionalidades comunes de las entidades, conteniendo la información necesaria de la tabla que representan.
- **Creador:** este patrón identifica quién será el responsable de la creación de nuevos objetos o clases. Un ejemplo de la utilización de este patrón es la clase Actions que propone el framework Symfony, en ella se encuentran las acciones definidas para el sistema, en ella se crean los objetos de las clases que representan las entidades.
- **Controlador:** este patrón plantea que la lógica de negocios debe estar separada de la capa de presentación para aumentar la reutilización de código y mayor control. En Symfony las peticiones web son manipuladas por solo dos controladores

---

frontales (app.php, app\_dev.php), que son el punto de entrada único de toda la aplicación en un entorno determinado.

- **Alta Cohesión:** este patrón sugiere que toda la información que contenga una clase debe ser coherente y relacionada con la misma. Symfony ofrece una alta cohesión, sus clases constan de funcionalidades estrechamente relacionadas, definen las acciones para colaborar con otras, instanciar objetos y acceder a las properties.
- **Bajo Acoplamiento:** es el patrón que propone que las clases de una aplicación tengan el mayor grado de independencia posible, previendo que si hay modificaciones en una no se vean afectadas las demás. Las clases de Symfony que implementan la lógica del negocio y de acceso a datos se encuentran en el modelo, las cuales no tienen asociaciones con las de la vista o el controlador, lo que proporciona que la dependencia en este caso sea baja.

### **Patrones Gang of Four (GOF)**

Los patrones de diseño GOF, se clasifican en 3 categorías: creacionales, estructurales y de comportamiento.

**Creacionales:** tratan con las formas de crear instancias de objetos. El objetivo de estos patrones es de abstraer el proceso de instanciación y ocultar los detalles de cómo los objetos son creados o inicializados.

- **Object Pool (Conjunto de Objetos):** Se obtienen objetos nuevos a través de la clonación. Utilizado cuando el costo de crear una clase es mayor que el de clonarla. Especialmente con objetos muy complejos. Se especifica un tipo de objeto a crear y se utiliza una interfaz del prototipo para crear un nuevo objeto por clonación. El proceso de clonación se inicia instanciando un tipo de objeto de la clase que queremos clonar.
- **Abstract Factory (Fábrica abstracta):** Permite trabajar con objetos de distintas familias de manera que las familias no se mezclen entre sí y haciendo transparente el tipo de familia concreta que se esté usando.
- **Builder (Constructor virtual):** Abstrae el proceso de creación de un objeto complejo, centralizando dicho proceso en un único punto.
- **Factory Method (Método de fabricación):** Centraliza en una clase constructora la creación de objetos de un subtipo de un tipo determinado, ocultando al usuario la casuística para elegir el subtipo que crear.

- 
- Prototype (Prototipo): Crea nuevos objetos clonándolos de una instancia ya existente.
  - Singleton (Instancia única): Garantiza la existencia de una única instancia para una clase y la creación de un mecanismo de acceso global a dicha instancia.

**Estructurales:** Los patrones estructurales describen como las clases y objetos pueden ser combinados para formar grandes estructuras y proporcionar nuevas funcionalidades. Estos objetos adicionales pueden ser incluso objetos simples u objetos compuestos.

- Adapter (Adaptador): Adapta una interfaz para que pueda ser utilizada por una clase que de otro modo no podría utilizarla.
- Bridge (Puente): Desacopla una abstracción de su implementación.
- Composite (Objeto compuesto): Permite tratar objetos compuestos como si de un simple se tratase.
- Decorator (Envoltorio): Añade funcionalidad a una clase dinámicamente.
- Facade (Fachada): Provee de una interfaz unificada simple para acceder a una interfaz o grupo de interfaces de un subsistema.
- Flyweight (Peso ligero): Reduce la redundancia cuando gran cantidad de objetos poseen idéntica información.
- Proxy: Mantiene un representante de un objeto.

**Comportamiento:** Los patrones de comportamiento ayudan a definir la comunicación e iteración entre los objetos de un sistema. El propósito de este patrón es reducir el acoplamiento entre los objetos.

- Chain of Responsibility (Cadena de responsabilidad): Permite establecer la línea que deben llevar los mensajes para que los objetos realicen la tarea indicada.
- Command (Orden): Encapsula una operación en un objeto, permitiendo ejecutar dicha operación sin necesidad de conocer el contenido de la misma.
- Interpreter (Intérprete): Dado un lenguaje, define una gramática para dicho lenguaje, así como las herramientas necesarias para interpretarlo.
- Iterator (Iterador): Permite realizar recorridos sobre objetos compuestos independientemente de la implementación de estos.
- Mediator (Mediador): Define un objeto que coordine la comunicación entre objetos de distintas clases, pero que funcionan como un conjunto.
- Memento (Recuerdo): Permite volver a estados anteriores del sistema.

- Observer (Observador): Define una dependencia de uno-a-muchos entre objetos, de forma que cuando un objeto cambie de estado se notifique y actualicen automáticamente todos los objetos que dependen de él.
- State(Estado): Permite que un objeto modifique su comportamiento cada vez que cambie su estado interno.
- Strategy (Estrategia): Permite disponer de varios métodos para resolver un problema y elegir cuál utilizar en tiempo de ejecución.
- Template Method (Método plantilla): Define en una operación el esqueleto de un algoritmo, delegando en las subclases algunos de sus pasos, esto permite que las subclases redefinan ciertos pasos de un algoritmo sin cambiar su estructura.
- Visitor (Visitante): Permite definir nuevas operaciones sobre una jerarquía de clases sin modificar las clases sobre las que opera. (43)

Para el diseño del sistema se usarán los patrones GOF creacionales (Abstract Factory, Singleton, Factory Method), estructurales (Facade, Proxy), comportamiento (Strategy, State, Command).

## 2.6 Planificación

Luego de la realización de las HU, la metodología XP propone como parte de la planificación del proyecto realizar la estimación de todos los esfuerzos necesarios para implementar dichas HU. Esta estimación se mide por puntos, que representan las semanas ideales de trabajo, basándose en las ideas de que en estas semanas los desarrolladores no tendrán ningún tipo de interrupción.

A continuación se muestra la estimación de esfuerzo para las HU definidas para realizar el sistema de apoyo la autogestión del conocimiento en los CASIE.

No.	Requisito	Estimación
1	Gestionar Roles	1
2	Autenticar Usuarios	1
3	Ver Perfiles de Usuarios	1
4	Gestionar Grupos	1
5	Gestionar Preguntas	1
6	Gestionar Cuestionarios	1
7	Gestionar Diagnósticos	1
8	Gestionar Materiales	1
9	Gestionar Tópicos	1
10	Gestionar Contenidos	1
11	Gestionar Orientaciones de Trabajo	1

12	Realizar Cuestionario	1
13	Gestionar Tareas	1
14	Realizar Tareas	1
15	Revisar Tareas Asignadas	1
16	Notificación de actividades	1
17	Asignar Niveles	2

**Tabla 8: Estimación de Esfuerzo por Historias de Usuario del Sistema**

### 2.6.1 Plan de iteraciones

Luego de definir las HU y estimar los esfuerzos necesarios para su implementación, se definirán las etapas o iteraciones que se establecerán para su desarrollo. Se decidió realizar 3 iteraciones.

En la iteración 1 se desarrollarán las HU que permiten el funcionamiento básico del sistema.

Por su complejidad se dejó para una iteración 2 la HU-11, que es la referente al mecanismo de Inteligencia Artificial para la asignación de niveles en el sistema.

Para la iteración 3 se dejaron aquellas HU que no comprometen el funcionamiento del sistema, pero le dan valor extra al mismo, como es la notificación de acciones.

A continuación se muestra el Plan de duración de las Iteraciones, en el que se ordenan las HU en la medida en que serán implementadas y su duración.

Iteración	Orden de la HU a implementar.	Duración total
1	Gestionar Roles Autenticar Usuarios Ver Perfiles de Usuario Gestionar Grupos Gestionar Preguntas Gestionar Cuestionarios Gestionar Diagnósticos Gestionar Materiales Gestionar Tópicos Gestionar Contenidos Gestionar Orientaciones de Trabajo Realizar Cuestionarios Gestionar Tareas Realizar Tareas Revisar Tareas Asignadas	8 semanas
2	Asignar Niveles	1 semanas
3	Notificación de actividades	1 semana

**Tabla 9: Plan de duración de las Iteraciones del sistema**

---

## 2.6.2 Plan de Entrega

El Plan de Entrega es otro artefacto definido por XP en el que se planificarán las fechas de entrega de cada iteración y los productos obtenidos.

A continuación se muestra el Plan de entrega para cada iteración del sistema.

<b>1ra Entrega (Iteración 1) 15 de mayo de 2011</b>	<b>2da Entrega (Iteración 2) 22 de mayo de 2011</b>	<b>3ra Entrega (Iteración 3) 29 de mayo de 2011</b>
HU Gestionar Roles HU Autenticar Usuarios HU Ver Perfiles de Usuarios HU Gestionar Grupos HU Gestionar Preguntas HU Gestionar Cuestionarios HU Gestionar Diagnósticos HU Gestionar Materiales HU Gestionar Tópicos HU Gestionar Contenidos HU Gestionar Orientaciones de Trabajo HU Realizar Cuestionario HU Gestionar Tareas HU Realizar Tareas HU Revisar Tareas Asignadas	HU Notificación de actividades	HU Asignar Niveles

**Tabla 10: Plan de Entrega del Sistema**

## 2.7 Conclusiones parciales

En el capítulo se abordaron las características generales del sistema y los procesos a automatizar, además surgieron los artefactos según la metodología para desarrollar el mismo.

1. Luego de analizar los objetos de automatización y plantear la descripción del sistema, se realizó un modelo de dominio debido a que no están claramente identificados los procesos para el desarrollo del sistema. En la realización del mismo se identificaron las clases: Profesor, Administrador, Estudiante, Usuario, Niveles, Contenidos, Tareas, Cuestionarios, Preguntas, Materiales.
2. Se plantearon las Historias de Usuario con información básica de las necesidades del sistema, y se clasifican según su necesidad para su funcionamiento. Se agruparon 17 HU equivalentes a 17 requisitos funcionales para el sistema, de ellas

---

15 presentan alta prioridad, 1 prioridad media y 1 fue identificada como muy alta por su grado de dificultad.

3. Según las características apreciadas en la descripción del sistema se seleccionaron las herramientas, metodología y tecnologías adecuadas para el desarrollo de LanGlish:

- Metodología de desarrollo de software: XP.
- Lenguajes de Programación: PHP 5 y JavaScript 1.1, con el framework para PHP Symfony 2.0.
- Entorno de Desarrollo Integrado: NetBeans IDE 7.1.
- Sistema de Gestión de Base de Datos: PostgreSQL 8.4.
- Servidor Web: Apache 2.0.

4. Se realizó la descripción de la arquitectura del sistema y los patrones que se utilizarán. Se decidió usar Arquitectura en 3 Niveles, como patrón arquitectónico el Modelo Vista Controlador para separar la lógica del negocio, de la presentación y de los datos. Se analizaron también aquellos patrones de diseño que pueden ser usados para solucionar problemas comunes en el diseño, se identificaron:

- Patrones de Asignación de Responsabilidades: Experto, Creador, Controlador, Alta Cohesión, Bajo Acoplamiento.
- Patrones GOF: Abstract Factory, Singleton, Factory Method, Facade, Proxy, Strategy, Mediator, Command.

5. Como parte de la planificación del proyecto se estimaron los esfuerzos necesarios y el tiempo necesario para desarrollar las HU identificadas en 10 semanas de desarrollo con un Plan de Entrega de 3 iteraciones en los plazos: 15 de mayo de 2011, 22 de mayo de 2011 y 29 de mayo de 2011.

---

## Capítulo III: Iteración y Producción

En este capítulo se abordarán aspectos de la implementación y se definen las pruebas a realizar en el sistema. Se construirá un modelo de datos donde se especifican las entidades definidas para el diseño de la base de datos, elaboración de tareas de la ingeniería y tarjetas CRC, así como el proceso de realización de prueba. Las pruebas tienen como objetivo principal verificar si el software se adecua a los requisitos o a las condiciones necesarias para el cliente.

### 3.1 Tareas de Ingeniería

Las Tareas de Ingeniería se utilizan para describir brevemente las tareas que se realizarán durante el desarrollo del software. Pueden ser clasificadas en tareas de desarrollo, de corrección y mejora.

Estas tareas tienen relación con una historia de usuario; se especifica la fecha de inicio y fin de la tarea, se nombra al programador responsable de cumplirla y se describe que se tratará de hacer en la tarea. (44)

A continuación se muestran las Tareas de la Ingeniería para las HU de más alta complejidad, las demás pueden ser consultadas en el Anexo 4 de este documento.

Tarea de ingeniería	
<b>Número tarea:</b> 5	<b>Número historia de usuario:</b> HU-3
<b>Nombre tarea:</b> Gestionar Preguntas	
<b>Tipo de tarea:</b> Desarrollo	<b>Puntos estimados:</b> 1
<b>Fecha inicio:</b> 20/4/2012	<b>Fecha fin:</b> 24/4/2012
<b>Programador responsable:</b> Elayne Yinet Elejalde Román, Dariela Ferreiro Cabrera	
<b>Descripción:</b> desarrollar las funcionalidades de crear, modificar, eliminar y ver preguntas.	

Tabla 11: Tarea de la ingeniería de la tarea Gestionar Preguntas.

Tarea de ingeniería	
<b>Número tarea:</b> 6	<b>Número historia de usuario:</b> HU-4
<b>Nombre tarea:</b> Gestionar Cuestionarios	
<b>Tipo de tarea:</b> Desarrollo	<b>Puntos estimados:</b> 1

<b>Fecha inicio:</b> 20/4/2012	<b>Fecha fin:</b> 24/4/2012
<b>Programador responsable:</b> Dariela Ferreiro Cabrera, Elayne Yinet Elejalde Román	
<b>Descripción:</b> desarrollar las funcionalidades de crear, modificar, eliminar y ver cuestionarios.	

**Tabla 12: Tarea de la ingeniería de la tarea Gestionar Cuestionario.**

<b>Tarea de ingeniería</b>	
<b>Número tarea:</b> 7	<b>Número historia de usuario:</b> HU-5
<b>Nombre tarea:</b> Gestionar Diagnósticos	
<b>Tipo de tarea:</b> Desarrollo	<b>Puntos estimados:</b> 1
<b>Fecha inicio:</b> 24/4/2012	<b>Fecha fin:</b> 28/4/2012
<b>Programador responsable:</b> Elayne Yinet Elejalde Román, Dariela Ferreiro Cabrera	
<b>Descripción:</b> desarrollar las funcionalidades de crear, modificar, eliminar y ver diagnósticos.	

**Tabla 13: Tarea de la ingeniería de la tarea Gestionar Diagnóstico.**

<b>Tarea de ingeniería</b>	
<b>Número tarea:</b> 8	<b>Número historia de usuario:</b> HU-7
<b>Nombre tarea:</b> Gestionar Materiales	
<b>Tipo de tarea:</b> Desarrollo	<b>Puntos estimados:</b> 1
<b>Fecha inicio:</b> 24/4/2012	<b>Fecha fin:</b> 28/4/2012
<b>Programador responsable:</b> Dariela Ferreiro Cabrera, Elayne Yinet Elejalde Román	
<b>Descripción:</b> desarrollar las funcionalidades de adjuntar, modificar, eliminar y ver materiales.	

**Tabla 14: Tarea de la ingeniería de la tarea Gestionar Materiales.**

<b>Tarea de ingeniería</b>	
<b>Número tarea:</b> 17	<b>Número historia de usuario:</b> HU-11
<b>Nombre tarea:</b> Asignar Niveles	
<b>Tipo de tarea:</b> Desarrollo	<b>Puntos estimados:</b> 2
<b>Fecha inicio:</b> 22/5/2012	<b>Fecha fin:</b> 29/5/2012
<b>Programador responsable:</b> Dariela Ferreiro Cabrera, Elayne Yinet Elejalde Román	

**Descripción:** desarrollar las funcionalidades de asignación de niveles a los usuarios que realicen diagnósticos.

**Tabla 15: Tarea de la ingeniería de la tarea Asignar Niveles.**

### 3.2 Tarjetas Clase – Responsabilidad – Colaborador

Las tarjetas Clase-Responsabilidad-Colaborador (CRC) son artefactos que propone XP para organizar las clases más relevantes que implementen las principales funcionalidades del sistema.

Las tarjetas CRC constan de tres componentes fundamentales: la clase, que indica cómo se llamará la clase a implementar; la responsabilidad, donde se plantean todas las funcionalidades que debe hacer la clase; el colaborador, donde se mencionan otras clases involucradas y que son necesarias en algún momento para realizar las acciones.

A continuación se muestran las tarjetas CRC para las HU de más complejidad. Las demás serán ubicadas en el Anexo 5 de este documento.

<b>Clase Pregunta</b>	
<b>Descripción:</b> clase encargada de gestionar todo el proceso de preguntas	
<b>Responsabilidad</b>	<b>Colaborador</b>
Crear una pregunta	Nivel
Modificar una pregunta	
Eliminar una pregunta	
Ver una pregunta	

**Tabla 16: Tarjeta CRC de la clase Pregunta.**

<b>Clase Cuestionario</b>	
<b>Descripción:</b> clase encargada de gestionar todo el proceso referente a los cuestionarios	
<b>Responsabilidad</b>	<b>Colaborador</b>
Crear un cuestionario	Pregunta, Nivel
Modificar un cuestionario	
Eliminar un cuestionario	
Ver un cuestionario	
Realizar evaluación de cuestionarios enviados	

**Tabla 17: Tarjeta CRC de la clase Cuestionario.**

<b>Clase Materiales</b>	
<b>Descripción:</b> clase encargada de gestionar todo el proceso referente a los materiales	
<b>Responsabilidad</b>	<b>Colaborador</b>
Adjuntar un material	Tarea, Niveles
Modificar un material	
Eliminar un material	
Ver un material	

**Tabla 18: Tarjeta CRC de la clase Materiales.**

<b>Clase Diagnóstico</b>	
<b>Descripción:</b> clase encargada de gestionar todo el proceso referente a los diagnósticos	
<b>Responsabilidad</b>	<b>Colaborador</b>
Crear un diagnóstico	Pregunta, Nivel
Modificar un diagnóstico	
Eliminar un diagnóstico	
Ver un diagnóstico	

**Tabla 19: Tarjeta CRC de la clase Diagnóstico.**

<b>Clase Asignar Niveles</b>	
<b>Descripción:</b> clase encargada de asignar niveles.	
<b>Responsabilidad</b>	<b>Colaborador</b>
Asignar nivel al estudiante	Cuestionario

**Tabla 20: Tarjeta CRC de la clase Asignar Niveles.**

### 3.3 Modelo de Datos

El modelo de datos permite representar los contenidos en la base de datos del sistema y sus relaciones. Proporciona una abstracción de los datos y métodos que facilitan la implementación del sistema. A continuación se muestra el modelo de datos del sistema:

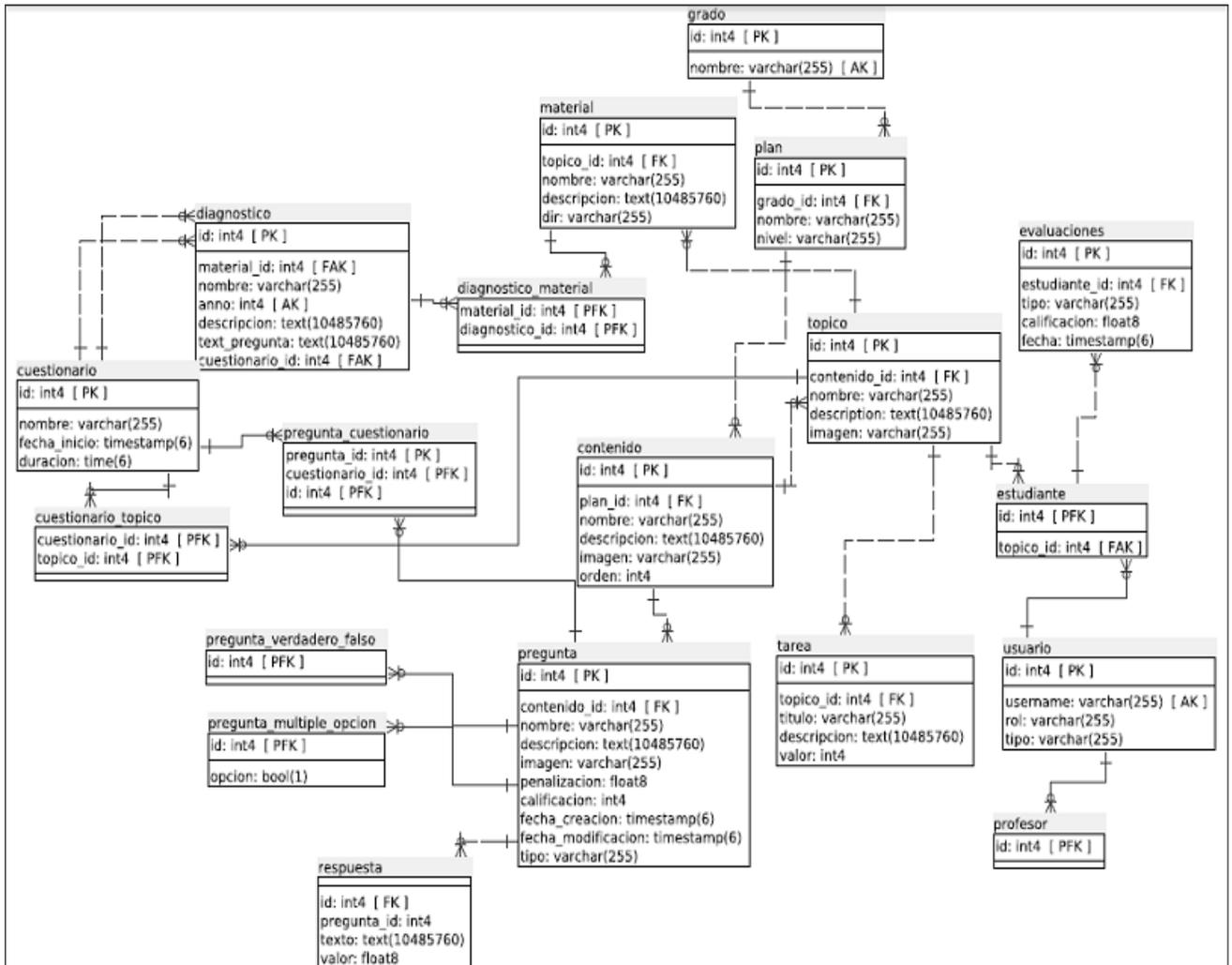
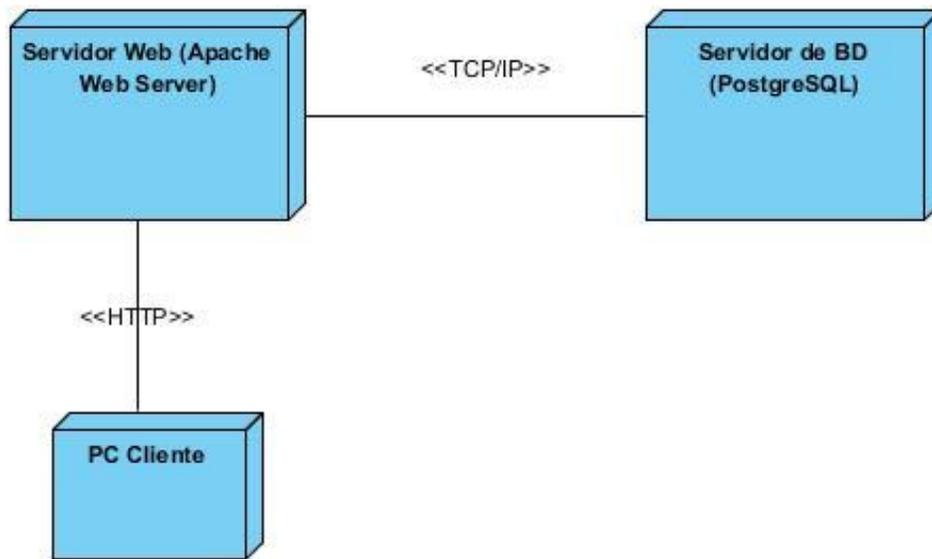


Figura 3: Modelo de datos del sistema.

### 3.4 Modelo de Despliegue

Los Diagramas de Despliegue son usados para mostrar la arquitectura física de un sistema, permite analizar la configuración de los elementos de software, hardware, procesos y objetos que se ejecutan en forma de nodos físicos.

A continuación se muestra la distribución de nodos físicos del sistema:



**Figura 4: Modelo de Despliegue del sistema.**

### 3.5 Validación del Sistema

En vista de lograr un producto con la calidad requerida se debe confeccionar un plan de pruebas, donde en cada prueba se indicarán las posibles respuestas que tendrá el software en la ejecución de cada funcionalidad, así como los posibles mensajes de error o información emitidos por el mismo cuando se estén probando dichas funcionalidades.

El proceso de pruebas se dirige al sistema de software en general, el mismo tiene como objetivo medir hasta donde el software cumple con los requisitos determinados por el cliente. Las pruebas de software validan y verifican la calidad del mismo. Son utilizadas para detectar posibles errores surgidos durante el desarrollo del producto, colocando en riesgo la usabilidad o la calidad del mismo.

La metodología XP gestiona los procesos de prueba con pocos artefactos, de forma tal que la interrelación de los desarrolladores y el cliente, logren la obtención del producto deseado.

La metodología XP divide las pruebas del sistema en dos grupos:

**Pruebas Unitarias:** Son las encargadas de verificar el código de forma automática y son diseñadas por los programadores. Estas pruebas fueron realizadas cada vez que se termina de implementar alguna funcionalidad probándola directamente en el entorno real. Las pruebas unitarias al sistema fueron provistas y generadas por el framework de desarrollo Symfony. Todas las pruebas unitarias que proporciona Symfony se ubican en el

---

directorio test/ del proyecto, bajo el subdirectorio test/unit/. Este framework permite y recomienda las pruebas unitarias proveyendo al sistema de un conjunto de herramientas con este fin, como es la librería Lime. Lime ejecuta métodos y funciones con conjuntos predefinidos de datos y argumentos, mostrando mensajes de error ante cualquier fallo y comparaciones entre las salidas y los resultados esperados para determinar si la prueba procede; por lo que para la ejecución de este tipo de pruebas se puso en práctica la librería Lime del Framework Symfony, la cual está altamente calificada para la realización de pruebas unitarias

**Pruebas de Aceptación:** También llamadas pruebas del cliente, se enfocan en las funcionalidades del sistema, es decir son un conjunto de condiciones utilizadas para determinar si un requisito es parcial o completamente satisfactorio. Por lo que las mismas se recogerán en el documento. Fueron realizadas por el cliente con la utilización de Casos de prueba, los mismos se realizarán teniendo en cuenta el orden y la prioridad que se le ha sido asignada a las Historias de Usuarios.

Los Casos de Pruebas correspondientes a cada una de las funcionalidades serán descritos mediante tablas divididas por las siguientes secciones:

- **Código Caso de Prueba:** Contiene el identificador de caso de prueba (en el caso de las presentes, se utiliza el identificador de la HU, al que se le adiciona ‘\_P’ y un número consecutivo, HU-X-P#).
- **Nombre Historia de Usuario:** Contiene el nombre de la HU correspondiente a este caso de prueba.
- **Nombre de la persona que realiza la prueba:** Contiene el nombre del responsable que realiza la prueba.
- **Descripción de la Prueba:** Contiene una breve descripción de la prueba realizada.
- **Condiciones de Ejecución:** Se incluyen las condiciones necesarias para que se pueda realizar la prueba.
- **Entrada/Pasos de ejecución:** Contiene una serie de pasos enumerados para lograr realizar la prueba de esta HU.
- **Resultado Esperado:** Contiene la descripción de lo que se espera luego de realizar la prueba (cumplimiento de las restricciones del producto).
- **Evaluación de la Prueba:** Muestra si la prueba fue satisfactoria o insatisfactoria.

Para la realización de las Pruebas de Aceptación se definieron casos de prueba para cada uno de los escenarios de todas las HU, mostrando a continuación las de prioridad más significativa (muy altas), y las de prioridad alta, media o baja podrán consultarse en el Anexo 6.

Casos de Prueba de Aceptación	
<b>Código de Caso Prueba:</b> HU-3-P1	<b>Nombre de Historia de Usuario:</b> Gestionar Preguntas.
<b>Nombre de la persona que realiza la prueba:</b> Maura Berta Hidalgo	
<b>Descripción de la Prueba:</b> Se insertarán los datos de una pregunta, inicialmente se probarán con datos incorrectos para verificar las validaciones del sistema, luego de forma correcta para comprobar que los datos sean almacenados.	
<b>Condiciones de Ejecución:</b> El usuario debe tener los permisos suficientes para realizar esta operación.	
<b>Entrada/Pasos de ejecución:</b> <ol style="list-style-type: none"> <li>1. Insertar los datos en blanco.</li> <li>2. Insertar los datos (con un campo vacío, hacer todas las combinaciones)</li> <li>3. Insertar un nombre que ya esté en la base de datos.</li> <li>4. Insertar los datos (un campo con datos incorrectos, hacer todas las combinaciones)</li> <li>5. Insertar los datos correctamente.</li> <li>6. Verificar que la pregunta aparezca en el listado.</li> </ol>	
<b>Resultado Esperado:</b> El sistema debe alertar al usuario cuando se inserten datos en blanco en los campos obligatorios (contenido, nombre, descripción, penalización, calificación, fecha de creación, fecha de modificación) o existan datos incorrectos; además debe alertar cuando ya exista un elemento con el mismo nombre, almacenado en la base de datos. Cuando se inserten los datos correctamente, el sistema debe almacenarlos en la base de datos.	
<b>Evaluación de la Prueba:</b> Evaluación satisfactoria	

**Tabla 21: Caso de Prueba 1 de HU Gestionar Preguntas.**

Casos de Prueba de Aceptación	
<b>Código de Caso Prueba:</b> HU-4-P2	<b>Nombre de Historia de Usuario:</b> Gestionar Preguntas.
<b>Nombre de la persona que realiza la prueba:</b> Maura Berta Hidalgo	
<b>Descripción de la Prueba:</b> Ver los datos que contiene una pregunta.	
<b>Condiciones de Ejecución:</b> El usuario debe tener los permisos suficientes para realizar esta operación.	
<b>Entrada/Pasos de ejecución:</b> <ol style="list-style-type: none"> <li>1. Acceder al listado de preguntas.</li> <li>2. Seleccionar una pregunta para ver los detalles de la misma.</li> <li>3. Seleccionar la opción Modificar (Ver Caso de Prueba: HU-3-P3).</li> <li>4. Seleccionar la opción Eliminar (Ver Caso de Prueba: HU-3-P4).</li> </ol>	
<b>Resultado Esperado:</b> El sistema debe mostrar los datos de la pregunta, así como poder realizar las acciones que permiten modificar o eliminar la pregunta.	

---

---

**Evaluación de la Prueba:** Evaluación satisfactoria

**Tabla 22: Caso de Prueba 2 de HU Gestionar Pregunta.**

Casos de Prueba de Aceptación	
<b>Código de Caso Prueba:</b> HU-3-P3	<b>Nombre de Historia de Usuario:</b> Gestionar Preguntas.
<b>Nombre de la persona que realiza la prueba:</b> Maura Berta Hidalgo	
<b>Descripción de la Prueba:</b> Se modifican los datos de una pregunta, inicialmente se modificarán incorrectamente para verificar las validaciones del sistema, luego de forma correcta para comprobar que los datos sean almacenados y cargados.	
<b>Condiciones de Ejecución:</b> El usuario debe tener los permisos suficientes para realizar esta operación.	
<b>Entrada/Pasos de ejecución:</b> <ol style="list-style-type: none"><li>1. Modificar los datos en los campos requeridos e insertarlos en blanco.</li><li>2. Modificar los datos en los campos requeridos e insertarlos (campo vacío).</li><li>3. Modificar el nombre e insertarlo estando ya en la base de datos.</li><li>4. Modificar los datos en los campos requeridos en insertarlos (datos incorrectos).</li><li>5. Modificar los datos correctamente.</li><li>6. Verificar que la pregunta aparezca en el listado con los datos actualizados.</li></ol>	
<b>Resultado Esperado:</b> El sistema debe alertar al usuario cuando se inserten datos en blanco en los campos obligatorios (contenido, nombre, descripción, penalización, calificación, fecha de creación, fecha de modificación) o existan datos incorrectos; además debe alertar cuando ya exista un elemento con el mismo nombre, almacenado en la base de datos. Cuando se modifiquen los datos correctamente, el sistema debe almacenarlos en la base de datos y mostrar los datos actualizados cuando se soliciten.	
<b>Evaluación de la Prueba:</b> Evaluación satisfactoria	

**Tabla 23: Caso de Prueba 3 de HU Gestionar Pregunta.**

Casos de Prueba de Aceptación	
<b>Código de Caso Prueba:</b> HU-3-P4	<b>Nombre de Historia de Usuario:</b> Gestionar Preguntas.
<b>Nombre de la persona que realiza la prueba:</b> Maura Berta Hidalgo	
<b>Descripción de la Prueba:</b> Se elimina una pregunta	
<b>Condiciones de Ejecución:</b> El usuario debe tener los permisos suficientes para realizar esta operación.	
<b>Entrada/Pasos de ejecución:</b> <ol style="list-style-type: none"><li>1. Seleccionar la opción Eliminar pregunta.</li><li>2. Verificar que la pregunta no aparezca en el listado.</li></ol>	
<b>Resultado Esperado:</b> El sistema debe mostrar una ventana de aviso al usuario preguntando si realmente desea eliminar. Cuando se acepte, debe eliminar la pregunta. Después de eliminado, el elemento no debe aparecer en la lista.	
<b>Evaluación de la Prueba:</b> Evaluación satisfactoria	

**Tabla 24: Caso de Prueba 4 de HU Gestionar Pregunta.**

Casos de Prueba de Aceptación	
<b>Código de Caso Prueba:</b>	<b>Nombre de Historia de Usuario:</b> Gestionar Cuestionarios.

HU-4-P1	
<b>Nombre de la persona que realiza la prueba:</b> Maura Berta Hidalgo	
<b>Descripción de la Prueba:</b> Se insertarán los datos de un cuestionario, inicialmente se insertarán incorrectamente para verificar las validaciones del sistema, luego de forma correcta para comprobar que los datos sean almacenados.	
<b>Condiciones de Ejecución:</b> El usuario debe tener los permisos suficientes para realizar esta operación.	
<b>Entrada/Pasos de ejecución:</b> <ol style="list-style-type: none"> <li>1. Insertar los datos en blanco.</li> <li>2. Insertar los datos (con un campo vacío, hacer todas las combinaciones)</li> <li>3. Insertar un nombre que ya esté en la base de datos.</li> <li>4. Insertar los datos (un campo con datos incorrectos, hacer todas las combinaciones)</li> <li>5. Insertar los datos correctamente.</li> <li>6. Verificar que el cuestionario aparezca en el listado.</li> </ol>	
<b>Resultado Esperado:</b> El sistema debe alertar al usuario cuando se inserten datos en blanco en los campos obligatorios (nombre, fecha de inicio, duración, tópico, pregunta) o existan datos incorrectos; además debe alertar cuando ya exista un elemento con el mismo nombre, almacenado en la base de datos. Cuando se inserten los datos correctamente, el sistema debe almacenarlos en la base de datos.	
<b>Evaluación de la Prueba:</b> Evaluación satisfactoria	

**Tabla 25: Caso de Prueba 1 de HU Gestionar Cuestionarios.**

Casos de Prueba de Aceptación	
<b>Código de Caso Prueba:</b> HU-4-P2	<b>Nombre de Historia de Usuario:</b> Gestionar Cuestionarios.
<b>Descripción de la Prueba:</b> Ver los datos que contiene un cuestionario.	
<b>Condiciones de Ejecución:</b> El usuario debe tener los permisos suficientes para realizar esta operación.	
<b>Entrada/Pasos de ejecución:</b> <ol style="list-style-type: none"> <li>1. Acceder al listado de cuestionarios.</li> <li>2. Seleccionar un cuestionario para ver los detalles del mismo.</li> <li>3. Seleccionar la opción Modificar (Ver Caso de Prueba: HU-4-P3).</li> <li>4. Seleccionar la opción Eliminar (Ver Caso de Prueba: HU-4-P4).</li> </ol>	
<b>Resultado Esperado:</b> El sistema debe mostrar los datos del cuestionario, así como poder realizar las acciones que permiten modificar o eliminar el cuestionario.	
<b>Evaluación de la Prueba:</b> Evaluación satisfactoria	

**Tabla 26: Caso de Prueba 2 de HU Gestionar Cuestionarios.**

Casos de Prueba de Aceptación	
<b>Código de Caso Prueba:</b> HU-4-P3	<b>Nombre de Historia de Usuario:</b> Gestionar Cuestionarios.
<b>Nombre de la persona que realiza la prueba:</b> Maura Berta Hidalgo	
<b>Descripción de la Prueba:</b> Se modifican los datos de un cuestionario, inicialmente se modificarán incorrectamente para verificar las validaciones del sistema, luego de forma correcta para comprobar que los datos sean almacenados y cargados.	
<b>Condiciones de Ejecución:</b> El usuario debe tener los permisos suficientes para realizar esta	

operación.
<b>Entrada/Pasos de ejecución:</b>
<ol style="list-style-type: none"> <li>1. Modificar los datos en los campos requeridos e insertarlos en blanco.</li> <li>2. Modificar los datos en los campos requeridos e insertarlos (campo vacío).</li> <li>3. Modificar el nombre e insertarlo estando ya en la base de datos.</li> <li>4. Modificar los datos en los campos requeridos en insertarlos (datos incorrectos).</li> <li>5. Modificar los datos correctamente.</li> <li>6. Verificar que el cuestionario aparezca en el listado con los datos actualizados.</li> </ol>
<b>Resultado Esperado:</b> El sistema debe alertar al usuario cuando se inserten datos en blanco en los campos obligatorios (nombre, fecha de inicio, duración, tópico, pregunta) o existan datos incorrectos; además debe alertar cuando ya exista un elemento con el mismo nombre, almacenado en la base de datos. Cuando se modifiquen los datos correctamente, el sistema debe almacenarlos en la base de datos y mostrar los datos actualizados cuando se soliciten.
<b>Evaluación de la Prueba:</b> Evaluación satisfactoria

**Tabla 27: Caso de Prueba 3 de HU Gestionar Cuestionarios.**

Casos de Prueba de Aceptación	
<b>Código de Caso Prueba:</b> HU-4-P4	<b>Nombre de Historia de Usuario:</b> Gestionar Cuestionarios.
<b>Nombre de la persona que realiza la prueba:</b> Maura Berta Hidalgo	
<b>Descripción de la Prueba:</b> Se elimina un cuestionario	
<b>Condiciones de Ejecución:</b> El usuario debe tener los permisos suficientes para realizar esta operación.	
<b>Entrada/Pasos de ejecución:</b>	
<ol style="list-style-type: none"> <li>1. Seleccionar la opción Eliminar cuestionario.</li> <li>2. Verificar que el cuestionario no aparezca en el listado.</li> </ol>	
<b>Resultado Esperado:</b> El sistema debe mostrar una ventana de aviso al usuario preguntando si realmente desea eliminar. Cuando se acepte, debe eliminar el cuestionario. Después de eliminado, el elemento no debe aparecer en la lista.	
<b>Evaluación de la Prueba:</b> Evaluación satisfactoria	

**Tabla 28: Caso de Prueba 4 de HU Gestionar Cuestionarios.**

Casos de Prueba de Aceptación	
<b>Código de Caso Prueba:</b> HU-5-P1	<b>Nombre de Historia de Usuario:</b> Gestionar Diagnósticos.
<b>Nombre de la persona que realiza la prueba:</b> Maura Berta Hidalgo	
<b>Descripción de la Prueba:</b> Se insertaran los datos de un diagnóstico, inicialmente se insertarán incorrectamente para verificar las validaciones del sistema, luego de forma correcta para comprobar que los datos sean almacenados.	
<b>Condiciones de Ejecución:</b> El usuario debe tener los permisos suficientes para realizar esta operación.	
<b>Entrada/Pasos de ejecución:</b>	
<ol style="list-style-type: none"> <li>7. Insertar los datos en blanco.</li> <li>8. Insertar los datos (con un campo vacío, hacer todas las combinaciones)</li> <li>9. Insertar un nombre que ya esté en la base de datos.</li> <li>10. Insertar los datos (un campo con datos incorrectos, hacer todas las combinaciones)</li> </ol>	

<p>11. Insertar los datos correctamente.</p> <p>12. Verificar que el diagnóstico aparezca en el listado.</p>
<p><b>Resultado Esperado:</b> El sistema debe alertar al usuario cuando se inserten datos en blanco en los campos obligatorios (material, nombre, año, descripción, texto de pregunta, cuestionario) o existan datos incorrectos; además debe alertar cuando ya exista un elemento con el mismo nombre, almacenado en la base de datos. Cuando se inserten los datos correctamente, el sistema debe almacenarlos en la base de datos.</p>
<p><b>Evaluación de la Prueba:</b> Evaluación satisfactoria</p>

**Tabla 29: Caso de Prueba 1 de HU Gestionar Diagnósticos**

Casos de Prueba de Aceptación	
<p><b>Código de Caso Prueba:</b> HU-5-P2</p>	<p><b>Nombre de Historia de Usuario:</b> Gestionar Diagnósticos.</p>
<p><b>Descripción de la Prueba:</b> Ver los datos que contiene un diagnóstico.</p>	
<p><b>Condiciones de Ejecución:</b> El usuario debe tener los permisos suficientes para realizar esta operación.</p>	
<p><b>Entrada/Pasos de ejecución:</b></p> <ol style="list-style-type: none"> <li>5. Acceder al listado de diagnósticos.</li> <li>6. Seleccionar un diagnóstico para ver los detalles del mismo.</li> <li>7. Seleccionar la opción Modificar (Ver Caso de Prueba: HU-5-P3).</li> <li>8. Seleccionar la opción Eliminar (Ver Caso de Prueba: HU-5-P4).</li> </ol>	
<p><b>Resultado Esperado:</b> El sistema debe mostrar los datos del diagnóstico, así como poder realizar las acciones que permiten modificar o eliminar el diagnóstico.</p>	
<p><b>Evaluación de la Prueba:</b> Evaluación satisfactoria</p>	

**Tabla 30: Caso de Prueba 2 de HU Gestionar Diagnóstico**

Casos de Prueba de Aceptación	
<p><b>Código de Caso Prueba:</b> HU-5-P3</p>	<p><b>Nombre de Historia de Usuario:</b> Gestionar Diagnósticos.</p>
<p><b>Nombre de la persona que realiza la prueba:</b> Maura Berta Hidalgo</p>	
<p><b>Descripción de la Prueba:</b> Se modifican los datos de un diagnóstico, inicialmente se modificarán incorrectamente para verificar las validaciones del sistema, luego de forma correcta para comprobar que los datos sean almacenados y cargados.</p>	
<p><b>Condiciones de Ejecución:</b> El usuario debe tener los permisos suficientes para realizar esta operación.</p>	
<p><b>Entrada/Pasos de ejecución:</b></p> <ol style="list-style-type: none"> <li>7. Modificar los datos en los campos requeridos e insertarlos en blanco.</li> <li>8. Modificar los datos en los campos requeridos e insertarlos (campo vacío).</li> <li>9. Modificar el nombre e insertarlo estando ya en la base de datos.</li> <li>10. Modificar los datos en los campos requeridos en insertarlos (datos incorrectos).</li> <li>11. Modificar los datos correctamente.</li> <li>12. Verificar que el diagnóstico aparezca en el listado con los datos actualizados.</li> </ol>	
<p><b>Resultado Esperado:</b> El sistema debe alertar al usuario cuando se inserten datos en blanco en los campos obligatorios (material, nombre, año, descripción, texto de pregunta, cuestionario) o existan datos incorrectos; además debe alertar cuando ya exista un elemento con el mismo nombre, almacenado en la base de datos. Cuando se modifiquen los datos correctamente, el sistema debe</p>	

almacenarlos en la base de datos y mostrar los datos actualizados cuando se soliciten.
<b>Evaluación de la Prueba:</b> Evaluación satisfactoria

**Tabla 31: Prueba 3 de HU Gestionar Diagnóstico**

Casos de Prueba de Aceptación	
<b>Código de Caso Prueba:</b> HU-5-P4	<b>Nombre de Historia de Usuario:</b> Gestionar Diagnósticos.
<b>Nombre de la persona que realiza la prueba:</b> Maura Berta Hidalgo	
<b>Descripción de la Prueba:</b> Se elimina un diagnóstico	
<b>Condiciones de Ejecución:</b> El usuario debe tener los permisos suficientes para realizar esta operación.	
<b>Entrada/Pasos de ejecución:</b> <ol style="list-style-type: none"> <li>3. Seleccionar la opción Eliminar diagnóstico.</li> <li>4. Verificar que el diagnóstico no aparezca en el listado.</li> </ol>	
<b>Resultado Esperado:</b> El sistema debe mostrar una ventana de aviso al usuario preguntando si realmente desea eliminar. Cuando se acepte, debe eliminar el diagnóstico. Después de eliminado, el elemento no debe aparecer en la lista.	
<b>Evaluación de la Prueba:</b> Evaluación satisfactoria	

**Tabla 32: Prueba 4 de HU Gestionar Diagnóstico**

Casos de Prueba de Aceptación	
<b>Código de Caso Prueba:</b> HU-7-P1	<b>Nombre de Historia de Usuario:</b> Gestionar Materiales.
<b>Nombre de la persona que realiza la prueba:</b> Maura Berta Hidalgo	
<b>Descripción de la Prueba:</b> Se insertarán los datos que debe tener un material así como su adjunto, inicialmente se insertarán incorrectamente para verificar las validaciones del sistema, luego de forma correcta para comprobar que los datos sean almacenados.	
<b>Condiciones de Ejecución:</b> El usuario debe tener los permisos suficientes para realizar esta operación.	
<b>Entrada/Pasos de ejecución:</b> <ol style="list-style-type: none"> <li>1. Insertar los datos en blanco sin adjunto.</li> <li>2. Insertar los datos (con un campo vacío, hacer todas las combinaciones)</li> <li>3. Insertar un nombre que ya esté en la base de datos.</li> <li>4. Insertar los datos (un campo con datos incorrectos, hacer todas las combinaciones)</li> <li>5. Insertar los datos correctamente sin adjunto.</li> <li>6. Insertar los datos correctamente con adjunto.</li> <li>7. Verificar que el material aparezca en el listado.</li> </ol>	
<b>Resultado Esperado:</b> El sistema debe alertar al usuario cuando se inserten datos en blanco en los campos obligatorios (tópico, nombre, descripción, dirección) o existan datos incorrectos; además debe alertar cuando ya exista un elemento con el mismo nombre, almacenado en la base de datos. Cuando se inserten los datos correctamente, el sistema debe almacenarlos en la base de datos.	
<b>Evaluación de la Prueba:</b> Evaluación satisfactoria	

**Tabla 33: Caso de Prueba 1 de HU Gestionar Materiales.**

Casos de Prueba de Aceptación	
<b>Código de Caso Prueba:</b>	<b>Nombre de Historia de Usuario:</b> Gestionar Materiales.

HU-7-P2	
<b>Nombre de la persona que realiza la prueba:</b> Maura Berta Hidalgo	
<b>Descripción de la Prueba:</b> Ver los datos que contiene un material.	
<b>Condiciones de Ejecución:</b> El usuario debe tener los permisos suficientes para realizar esta operación.	
<b>Entrada/Pasos de ejecución:</b>	
<ol style="list-style-type: none"> <li>1. Acceder al listado de materiales.</li> <li>2. Seleccionar un material para ver los detalles del mismo.</li> <li>3. Seleccionar la opción Modificar (Ver Caso de Prueba: HU-7-P3).</li> <li>4. Seleccionar la opción Eliminar (Ver Caso de Prueba: HU-7-P4).</li> </ol>	
<b>Resultado Esperado:</b> El sistema debe mostrar los datos del Material, así como poder realizar las acciones que permiten modificar o eliminar el Material.	
<b>Evaluación de la Prueba:</b> Evaluación satisfactoria	

**Tabla 34: Caso de Prueba 2 de HU Gestionar Materiales.**

Casos de Prueba de Aceptación	
<b>Código de Caso Prueba:</b> HU-7-P3	<b>Nombre de Historia de Usuario:</b> Gestionar Materiales.
<b>Nombre de la persona que realiza la prueba:</b> Maura Berta Hidalgo	
<b>Descripción de la Prueba:</b> Se modifican los datos de un material, inicialmente se modificarán incorrectamente para verificar las validaciones del sistema, luego de forma correcta para comprobar que los datos sean almacenados y cargados.	
<b>Condiciones de Ejecución:</b> El usuario debe tener los permisos suficientes para realizar esta operación.	
<b>Entrada/Pasos de ejecución:</b>	
<ol style="list-style-type: none"> <li>1. Modificar los datos en los campos requeridos e insertarlos en blanco.</li> <li>2. Modificar los datos en los campos requeridos e insertarlos (campo vacío).</li> <li>3. Modificar el nombre e insertarlo estando ya en la base de datos.</li> <li>4. Modificar los datos en los campos requeridos en insertarlos (datos incorrectos).</li> <li>5. Modificar los datos correctamente.</li> <li>6. Verificar que el material aparezca en el listado con los datos actualizados.</li> </ol>	
<b>Resultado Esperado:</b> El sistema debe alertar al usuario cuando se inserten datos en blanco en los campos obligatorios (tópico, nombre, descripción, dirección) o existan datos incorrectos; además debe alertar cuando ya exista un elemento con el mismo nombre, almacenado en la base de datos. Cuando se modifiquen los datos correctamente, el sistema debe almacenarlos en la base de datos y mostrar los datos actualizados cuando se soliciten.	
<b>Evaluación de la Prueba:</b> Evaluación satisfactoria	

**Tabla 35: Caso de Prueba 3 de HU Gestionar Materiales.**

Casos de Prueba de Aceptación	
<b>Código de Caso Prueba:</b> HU-7-P4	<b>Nombre de Historia de Usuario:</b> Gestionar Materiales.
<b>Nombre de la persona que realiza la prueba:</b> Maura Berta Hidalgo	
<b>Descripción de la Prueba:</b> Se elimina un material	
<b>Condiciones de Ejecución:</b> El usuario debe tener los permisos suficientes para realizar esta operación.	

<b>Entrada/Pasos de ejecución:</b>
<ol style="list-style-type: none"> <li>1. Seleccionar la opción Eliminar material.</li> <li>2. Verificar que el material no aparezca en el listado.</li> </ol>
<b>Resultado Esperado:</b> El sistema debe mostrar una ventana de aviso al usuario preguntando si realmente desea eliminar. Cuando se acepte, debe eliminar el material. Después de eliminado, el elemento no debe aparecer en la lista.
<b>Evaluación de la Prueba:</b> Evaluación satisfactoria

**Tabla 36: Caso de Prueba 4 de HU Gestionar Materiales.**

Casos de Prueba de Aceptación	
<b>Código de Caso Prueba:</b> HU-11-P1	<b>Nombre de Historia de Usuario:</b> Asignar Niveles.
<b>Nombre de la persona que realiza la prueba:</b> Maura Berta Hidalgo	
<b>Descripción de la Prueba:</b> Se verificará que después de realizado el diagnóstico inicial (Ver Caso de Prueba: HU-6-P1), el sistema ubique al estudiante en un grupo según el nivel que se le fue asignado.	
<b>Condiciones de Ejecución:</b> Debe haberse realizado el diagnóstico inicial.	
<b>Entrada/Pasos de ejecución:</b>	
<ol style="list-style-type: none"> <li>1. Buscar al estudiante que se utilizó como prueba para realizar el diagnóstico inicial.</li> <li>2. Verificar si aparece con un nivel asignado, de ser así verificar si el estudiante está ubicado en el grupo que le corresponde, según el nivel y el año que cursa.</li> </ol>	
<b>Resultado Esperado:</b> El sistema debe ubicar al estudiante según el nivel asignado y el año que cursa, en el grupo que le corresponde.	
<b>Evaluación de la Prueba:</b> Evaluación satisfactoria	

**Tabla 37: Caso de Prueba 1 de HU Asignar Niveles.**

Las pruebas realizadas, permitieron elaborar un informe de las No Conformidades encontradas, para ser corregidas posteriormente. A continuación se muestran las tablas con la documentación recopilada durante las pruebas de los CP más significativos, los demás pueden ser consultados en el Anexo 7 de este documento.

Nombre de Historia de Usuario: Gestionar Preguntas		
Código de Caso Prueba: HU-3-P1		
Caminos de Ejecución	Resultados observables	NC
Enviar los campos en blanco.	No muestra la lista con la pregunta insertada.	ninguna
Insertar los datos dejando campos vacíos	Muestra un mensaje de información. No muestra la lista con la pregunta insertada.	ninguna
Insertar un nombre que ya esté en la base de datos.	Muestra un mensaje de información. No muestra la lista con la pregunta insertada.	funcionalidad
Llenar los campos con datos incorrectos.	Muestra un mensaje de información. No muestra la lista con la pregunta insertada.	ninguna
Insertar los datos Correctamente.	Muestra la lista con la pregunta	ninguna

	insertada.	
<b>Código de Caso Prueba: HU-3-P2</b>		
Seleccionar una pregunta y ver los detalles de la misma.	Muestra la pregunta con todos sus datos.	ninguna
Seleccionar la opción modificar.	Muestra la interfaz de modificar.	ninguna
Seleccionar la opción eliminar.	Muestra la interfaz de eliminar.	ninguna
<b>Código de Caso Prueba: HU-3-P3</b>		
Enviar los campos en blanco.	No muestra la lista con la pregunta actualizada.	ninguna
Insertar los datos dejando campos vacíos	Muestra un mensaje de información. No muestra la lista con la pregunta actualizada.	ninguna
Insertar un nombre que ya esté en la base de datos.	Muestra un mensaje de información. No muestra la lista con la pregunta actualizada.	funcionalidad
Llenar los campos con datos incorrectos.	Muestra un mensaje de información. No muestra la lista con la pregunta actualizada.	ninguna
Modificar los datos Correctamente.	Muestra la lista con la pregunta actualizada.	ninguna
<b>Código de Caso Prueba: HU-3-P4</b>		
Seleccionar la opción de eliminar la pregunta.	Muestra la interfaz de eliminar.	ninguna
Se elimina la pregunta.	Muestra un mensaje de información. No muestra la lista con la pregunta eliminada.	ninguna
Se elimina la pregunta.	Muestra la lista con la pregunta eliminada.	ninguna

**Tabla 38: Resultados de las pruebas de HU Gestionar Preguntas.**

<b>Nombre de Historia de Usuario: Gestionar Cuestionarios</b>		
<b>Código de Caso Prueba: HU-4-P1</b>		
<b>Caminos de Ejecución</b>	<b>Resultados observables</b>	<b>NC</b>
Enviar los campos en blanco.	No muestra la lista con el cuestionario insertado.	ninguna
Insertar los datos dejando campos vacíos	Muestra un mensaje de información. No muestra la lista con el cuestionario insertado.	ninguna
Insertar un nombre que ya esté en la base de datos.	Muestra un mensaje de información. No muestra la lista con el cuestionario insertado.	funcionalidad
Llenar los campos con datos incorrectos.	Muestra un mensaje de información. No muestra la lista con el cuestionario insertado.	ninguna
Insertar los datos Correctamente.	Muestra la lista con el cuestionario insertado.	ninguna
<b>Código de Caso Prueba: HU-4-P2</b>		

Seleccionar un cuestionario y ver los detalles del mismo.	Muestra el cuestionario con todos sus datos.	ninguna
Seleccionar la opción modificar.	Muestra la interfaz de modificar.	ninguna
Seleccionar la opción eliminar.	Muestra la interfaz de eliminar.	ninguna
<b>Código de Caso Prueba: HU-4-P3</b>		
Enviar los campos en blanco.	No muestra la lista con el cuestionario actualizado.	ninguna
Insertar los datos dejando campos vacíos	Muestra un mensaje de información. No muestra la lista con el cuestionario actualizado.	ninguna
Insertar un nombre que ya esté en la base de datos.	Muestra un mensaje de información. No muestra la lista con el cuestionario actualizado.	funcionalidad
Llenar los campos con datos incorrectos.	Muestra un mensaje de información. No muestra la lista con el cuestionario actualizado.	ninguna
Modificar los datos Correctamente.	Muestra la lista con el cuestionario actualizado.	ninguna
<b>Código de Caso Prueba: HU-4-P4</b>		
Seleccionar la opción de eliminar cuestionario.	Muestra la interfaz de eliminar.	ninguna
Se elimina el cuestionario.	Muestra un mensaje de información. No muestra la lista con el cuestionario eliminado.	ninguna
Se elimina el cuestionario.	Muestra la lista con el cuestionario eliminado.	ninguna

**Tabla 39: Resultados de las pruebas de HU Gestionar Cuestionarios.**

<b>Nombre de Historia de Usuario: Gestionar Diagnósticos</b>		
<b>Código de Caso Prueba: HU-5-P1</b>		
<b>Caminos de Ejecución</b>	<b>Resultados observables</b>	<b>NC</b>
Enviar los campos en blanco.	No muestra la lista con el diagnóstico insertado.	ninguna
Insertar los datos dejando campos vacíos	Muestra un mensaje de información. No muestra la lista con el diagnóstico insertado.	ninguna
Insertar un nombre que ya esté en la base de datos.	Muestra un mensaje de información. No muestra la lista con el diagnóstico insertado.	funcionalidad
Llenar los campos con datos incorrectos.	Muestra un mensaje de información. No muestra la lista con el diagnóstico insertado.	ninguna
Insertar los datos Correctamente.	Muestra la lista con el diagnóstico insertado.	ninguna
<b>Código de Caso Prueba: HU-5-P2</b>		
Seleccionar un diagnóstico y ver los detalles del mismo.	Muestra el diagnóstico con todos sus datos.	ninguna

Seleccionar la opción modificar.	Muestra la interfaz de modificar.	ninguna
Seleccionar la opción eliminar.	Muestra la interfaz de eliminar.	ninguna
<b>Código de Caso Prueba: HU-5-P3</b>		
Enviar los campos en blanco.	No muestra la lista con el diagnóstico actualizado.	ninguna
Insertar los datos dejando campos vacíos	Muestra un mensaje de información. No muestra la lista con el diagnóstico actualizado.	ninguna
Insertar un nombre que ya esté en la base de datos.	Muestra un mensaje de información. No muestra la lista con el diagnóstico actualizado.	funcionalidad
Llenar los campos con datos incorrectos.	Muestra un mensaje de información. No muestra la lista con el diagnóstico actualizado.	ninguna
Modificar los datos Correctamente.	Muestra la lista con el diagnóstico actualizado.	ninguna
<b>Código de Caso Prueba: HU-5-P4</b>		
Seleccionar la opción de eliminar diagnóstico.	Muestra la interfaz de eliminar.	ninguna
Se elimina el diagnóstico.	Muestra un mensaje de información. No muestra la lista con el diagnóstico eliminado.	ninguna
Se elimina el diagnóstico.	Muestra la lista con el diagnóstico eliminado.	ninguna

**Tabla 40: Resultados de las pruebas de HU Gestionar Diagnósticos.**

<b>Nombre de Historia de Usuario: Gestionar Materiales</b>		
<b>Código de Caso Prueba: HU-7-P1</b>		
<b>Caminos de Ejecución</b>	<b>Resultados observables</b>	<b>NC</b>
Enviar los campos en blanco.	No muestra la lista con el material insertado.	ninguna
Insertar los datos dejando campos vacíos	Muestra un mensaje de información. No muestra la lista con el material insertado.	ninguna
Insertar un nombre que ya esté en la base de datos.	Muestra un mensaje de información. No muestra la lista con el material insertado.	funcionalidad
Llenar los campos con datos incorrectos.	Muestra un mensaje de información. No muestra la lista con el material insertado.	ninguna
Insertar los datos Correctamente.	Muestra la lista con el material insertado.	ninguna
<b>Código de Caso Prueba: HU-7-P2</b>		
Seleccionar un material y ver los detalles del mismo.	Muestra el material con todos sus datos.	ninguna
Seleccionar la opción modificar.	Muestra la interfaz de modificar.	ninguna
Seleccionar la opción eliminar.	Muestra la interfaz de eliminar.	ninguna

<b>Código de Caso Prueba: HU-7-P3</b>		
Enviar los campos en blanco.	No muestra la lista con el material actualizado.	ninguna
Insertar los datos dejando campos vacíos	Muestra un mensaje de información. No muestra la lista con el material actualizado.	ninguna
Insertar un nombre que ya esté en la base de datos.	Muestra un mensaje de información. No muestra la lista con el material actualizado.	funcionalidad
Llenar los campos con datos incorrectos.	Muestra un mensaje de información. No muestra la lista con el material actualizado.	ninguna
Modificar los datos Correctamente.	Muestra la lista con el material actualizado.	ninguna
<b>Código de Caso Prueba: HU-7-P4</b>		
Seleccionar la opción de eliminar diagnóstico.	Muestra la interfaz de eliminar.	ninguna
Se elimina el material.	Muestra un mensaje de información. No muestra la lista con el material eliminado.	ninguna
Se elimina el material.	Muestra la lista con el material eliminado.	ninguna

**Tabla 41: Resultados de las pruebas de HU Gestionar Materiales.**

<b>Nombre de Historia de Usuario: Asignar Niveles</b>		
<b>Código de Caso Prueba: HU-11-P1</b>		
<b>Caminos de Ejecución</b>	<b>Resultados observables</b>	<b>NC</b>
Se realiza el diagnóstico inicial.	Se muestra la interfaz del usuario ya ubicado en un nivel según su año y resultado del diagnóstico.	ninguna
El usuario debe ser de 2do, 3er o 4to año para asignarle un nivel.	Le muestra el diagnóstico inicial.	ninguna
El usuario no pertenece a 2do, 3er o 4to año.	Muestra una interfaz con un mensaje de información y un link para salir del sistema.	ninguna
El usuario realiza el diagnóstico inicial y es ubicado en un nivel que no está creado.	Muestra una interfaz con un mensaje de información y un link para salir del sistema.	funcionalidad
El usuario ya pertenece a un grupo pero no está creado.	Muestra una interfaz con un mensaje de información y un link para salir del sistema.	ninguna

**Tabla 42: Resultados de las pruebas de HU Asignar Niveles.**

---

### 3.6 Conclusiones parciales

En este capítulo se detallaron procesos realizados durante la implementación y validación del sistema, según las características y artefactos generados en la utilización de la metodología XP:

1. Se conformaron las tarjetas Clase-Responsabilidad-Colaborador con las informaciones referentes a las clases, sus funcionalidades y relaciones con otras clases.
2. El Modelo de Datos fue realizado a través de la representación de las entidades identificadas y sus relaciones para la generación de la base de datos del sistema. Se representaron 19 entidades y las relaciones entre ellas, que serán equivalentes a las tablas de la base de datos.
3. El Modelo de Despliegue también fue realizado para representar los nodos físicos distribuidos, se planteó la necesidad de un servidor web Apache que se comunique con un servidor de base de datos con la instalación de PostgreSQL, y además se comunique con la PC cliente.
4. Se le realizaron las pruebas unitarias y de aceptación al software obteniendo un listado de No Conformidades de baja relevancia y por lo general se obtuvieron salidas positivas.

---

## Conclusiones Generales

Luego de finalizar la investigación, desarrollo, y teniendo en cuenta los resultados obtenidos, se pueden obtener las siguientes conclusiones:

1. Con el uso de las TIC, el proceso de enseñanza-aprendizaje del idioma Inglés ha obtenido mejores resultados, impactando los estándares educativos y mejorando los resultados y rendimiento de los estudiantes en las evaluaciones de la asignatura Inglés.
2. El estudio de los sistemas vinculados al campo de acción afirman que estos contribuyen al aprendizaje del inglés, pero no garantiza que la autogestión del conocimiento en la especialidad en inglés.
3. Los sistemas y procesos de gestión del conocimiento no son generalizables, por lo que se deben diseñar y desarrollar según los problemas específicos de cada organización.
4. Se obtuvo como resultado una aplicación web desarrollada completamente con tecnologías y herramientas libres, que cumple con todos los requerimientos especificados y buenas prácticas de desarrollo de software.
5. Se realizaron pruebas unitarias y de aceptación al sistema obteniendo salidas positivas. Los resultados arrojados afirman que la propuesta de solución cumple con las expectativas del cliente, pues resuelve el problema planteado por el mismo.

## **Recomendaciones**

Después de cumplir los objetivos propuestos al inicio del trabajo de diploma se recomienda:

1. Desarrollar nuevas versiones del software que permitan seguir mejorando las funcionalidades y capacidades del mismo,
2. Investigar e introducir otros conceptos referentes a la autogestión del conocimiento que se puedan vincular al software.

## **Glosario de Términos**

## Referencias Bibliográficas

1. ONE.cu. [En línea] 2012.  
[http://www.one.cu/publicaciones/06turismoycomercio/TIC/TIC Uso y Acceso en Cuba.pdf](http://www.one.cu/publicaciones/06turismoycomercio/TIC/TIC_Uso_y_Acceso_en_Cuba.pdf).
2. radiorebelde.cu. [En línea] 2008.  
<http://www.radiorebelde.cu/noticias/educacion/educacion1-171008.html>.
3. **Paula, Dr. C. Israel Núñez.** La gestión de la información, el conocimiento, la inteligencia y el aprendizaje organizacional desde una perspectiva socio-psicológica. [En línea] 2004. [Citado el: 23 de 5 de 2012.]  
[http://bvs.sld.cu/revistas/aci/vol12\\_3\\_04/aci04304.htm](http://bvs.sld.cu/revistas/aci/vol12_3_04/aci04304.htm).
4. **Nonaka, Takeuchi &. 1995.** 1995.
5. **Mazo, I., y Ortiz de Frutos, E.** *KMAT como herramienta de análisis de gestión del conocimiento. Cluster del conocimiento.* 1998.
6. **Artiles Visval, L. y S. Artiles Visva.** *Gestión del conocimiento y cooperación internacional en el ámbito universitario. Curso-Taller. Gestión del conocimiento para la cooperación y movilización de recursos en ambiente académico.* Lima : s.n., 2005.
7. **ANDREU, R. & SIEBER, S.** *La gestión integral del conocimiento y del aprendizaje.* s.l. : Economía Industrial, 1999. no. 326, pp. 63-72..
8. **Estrada Sentí, V. y J. P.Febles Rodríguez.** *Gestión del conocimiento en la Educación Superior. En: Las Nuevas Tecnologías de la Información y la Comunicación aplicadas a la Educación Superior.* Villa Clara : Editorial Feijoo, 2002.
9. **Ms. Ing. Edwin Valencia Castillo.** Gestión del Conocimiento. [En línea] 2010. [Citado el: 23 de 5 de 2012.] <http://www.unc.edu.pe>.
10. **Nidya Prieto Chavéz. Diplomado.** *Autogestión del Conocimiento y Tecnología Educativa,* 2008

11. **Prusak, Thomas H. Davenport and Laurence.** *Working Knowledge.* s.l. : Harvard Business School Press, 1998.
12. **LANGEFORS, B.** *Teoría de los Sistemas de Información.* Buenos Aires : El Ateneo, 1976.
13. **Druker, Peter F. 1983.** *New Relatives.* 1983.
14. **CAPOTE MARRERO, B. y GONZÁLEZ MACHÍN, D.** La gestión de información como herramienta fundamental en el desarrollo de los centros toxicológicos. [En línea] [http://bvs.sld.cu/revistas/aci/vol11\\_2\\_03/aci030203.htm](http://bvs.sld.cu/revistas/aci/vol11_2_03/aci030203.htm)..
15. Alfinee.es. [En línea] [Citado el: 1 de 5 de 2012.] [http://www.mariapinto.es/alfinee.es/aprender\\_analizar/como.htm](http://www.mariapinto.es/alfinee.es/aprender_analizar/como.htm)
16. **Rodríguez, David.** *Modelos para la creación y gestión del conocimiento: una aproximación teórica.* 2006.
17. TED-ED. [En línea] [Citado el: 26 de 5 de 2012.] <http://education.ted.com/>.
18. Longman English Interactive. [En línea] Pearson Education, 2008. [Citado el: 21 de 5 de 2012.] <http://www.longmanenglishinteractive.com/>.
19. Khan Academy [En línea] [Citado el: 26 de 5 de 2012.] <http://www.khanacademy.org/>.
20. **Pérez, Javier Eguíluz.** *Introducción a JAVASCRIPT.* 2007.
21. **Chuck Musciano, Bill Kennedy.** *HTML, la guía completa.* 1999.
22. **Parihar, Mridula.** *La Biblia de ASP.NET.* Madrid : EDICIONES ANAYA MULTIMEDIA,, 2002.
23. **Andi Gutmans, Stig Sæther Bakken, Derick Rethans.** *PHP 5 Power Programming.* 2004.
24. **Fabien Potencier, François Zaninotto.** *Symfony, la guía definitiva.* 2008.
25. librosweb.es. [En línea] 2012. [http://www.librosweb.es/symfony\\_1\\_1/capitulo1/symfony\\_en\\_pocas\\_palabras.html](http://www.librosweb.es/symfony_1_1/capitulo1/symfony_en_pocas_palabras.html).

26. ZFdes.com. [En línea] 2010.  
<http://manual.zfdes.com/es/introduction.overview.html>.
27. **Alejandro Morales Meza.** ammeza.com. [En línea] 2012.  
<http://www.ammeza.com/category/netbeans/>.
28. Programación en castellano. [En línea] 2011.  
[http://www.programacion.com/articulo/eclipse\\_-\\_i\\_-\\_historia\\_y\\_toma\\_de\\_contacto\\_288](http://www.programacion.com/articulo/eclipse_-_i_-_historia_y_toma_de_contacto_288).
29. SISTEMAS GESTORE DE BASE DE DATOS. SGDB libres. [En línea] 2008.  
<http://admin-info.galeon.com/gestor.pdf>.
30. **Lourdes Veronica Pilco.** Estudio Comparativo entre Sistemas de Gestión de Bases de Datos Libres Firebird y MySQL en el desarrollo de aplicaciones web. [En línea] 2011.  
<http://dspace.esPOCH.edu.ec/bitstream/123456789/1534/1/18T00475.pdf>.
31. PostgreSQL Cuba. [En línea] 2012. <http://postgresql.uci.cu/node/63>.
32. Editum.org. [En línea] 2006. [www.editum.org/Que-Es-Un-Servidor-Web-p-401.html](http://www.editum.org/Que-Es-Un-Servidor-Web-p-401.html).
33. Definición de Servidor IIS. [En línea] 2012.  
<http://sauce.pntic.mec.es/crer0052/iis/definici.htm>.
34. Informática Aplicada a la Gestión Pública. Facultad Derecho UMU. [En línea] 2011.  
<http://www.um.es/docencia/barzana/IAGP/IAGP2-Methodologias-de-desarrollo.html>.
35. **Gallego, Juan Pablo Gómez.** Fu damentos de la metodología RUP. 2007.
36. Metodología XP Vs. Metodología Rup. [En línea] 2008.  
<http://metodologiapvsmetodologiarup.blogspot.com/>.
37. **Msc Ailin Orjuela, Msc Mauricio Rojas.** Las Metodologías de Desarrollo Ágil como una Oportunidad para la Ingeniería de Software Educativo. [En línea] 2008.  
[http://www2.unalmed.edu.co/~pruebasminas/index.php?option=com\\_docman&task=doc\\_view&gid=340&tmpl=component&format=raw&Itemid=285](http://www2.unalmed.edu.co/~pruebasminas/index.php?option=com_docman&task=doc_view&gid=340&tmpl=component&format=raw&Itemid=285).

38. **Enrich, Margarita Fernández.** *Crystal Methodologies*. Valencia : Laboratorio de Sistemas de Información, 2003.
39. EcuRed. [En línea] 2012. [http://www.ecured.cu/index.php/Visual\\_Paradigm](http://www.ecured.cu/index.php/Visual_Paradigm).
40. PostgreSQL. [En línea] 2012. <http://archives.postgresql.org/pgsql-es-fomento/2009-07/msg00000.php>.
41. Proactiva Calidad. [En línea] 2012. <http://www.proactiva-calidad.com/java/patrones/mvc.html>
42. **E. Gamma, R. Helm, R. Johnson, J. Vlissides.** *Design Patterns*. 1995.
43. Geek The Planet. [En línea] 2012. <http://geektheplanet.net/5462/patrones-gof.xhtml>.
44. La wiki de la Programación Extrema. [En línea] 2012. <http://programacion-extrema.wikispaces.com/7.+Artefactos>.
45. Edutec. Revista Electrónica de Tecnología Educativa. [En línea] 2006. <http://www.uib.es/depart/gte/gte/edutec-e/revelec20/zulma20.htm>.
46. Universidad Tecnológica Nacional. [En línea] 2012. <http://www.utn.edu.ar/secretarias/planeamiento/ingles.utn>.
47. Lenguajes de programacion.com. [En línea] 2009. <http://www.lenguajes-de-programacion.com/lenguajes-de-programacion.shtml>.

## Bibliografía

[http://www.eduteka.org/ICT\\_InformeEuropa.php.html](http://www.eduteka.org/ICT_InformeEuropa.php.html)

[http://ec.europa.eu/education/pdf/doc254\\_en.pdf](http://ec.europa.eu/education/pdf/doc254_en.pdf)

Apache. [En línea] <http://httpd.apache.org/>.

Sito oficial del proyecto Doctrine. [En línea] [Citado el: 09 de 01 de 2011.]  
<http://www.doctrine-project.org/>.

Doctrine. [En línea] <http://www.doctrine-project.org/projects/orm>.

IBM. *IBM Rational Unified Process*. [En línea] <http://www-01.ibm.com/software/awdtools/rup/>.

Grupo soluciones GSINNova. *Rational Unified Process*. [En línea]  
<http://www.rational.com.ar/herramientas/rup.html>.

PostgreSQL. *PostgreSQL Global Development Group*. [En línea]  
[http://www.postgresql.org/..](http://www.postgresql.org/)

PHP. [En línea] <http://www.php.net/manual/es/index.php..>

**Villar, Malay Rodríguez.** Introducción de procedimientos Ágiles en la producción de software en la Facultad 7 de la Universidad de las Ciencias Informáticas. [En línea]  
<http://bibliodoc.uci.cu/TD/T>.

**Grau Abalo, Ricardo, Correa Valdés, Cecilia y Rojas Betancur, Mauricio.** *METODOLOGÍA DE LA INVESTIGACIÓN*.

**Canós, José H., Letelier, Patricio y Penadés, M<sup>a</sup> Carmen.** *Métodologías Ágiles en el Desarrollo de Software*. Valencia : Universidad Politécnica de Valencia. : s.n.

**Fernández Escribano, Gerardo.** *Introducción a Extreme Programming*.

**Jacobson, Ivar, Booch, Grady y Rumbaugh, James.** *El proceso unificado de desarrollo de software*. Madrid : Pearson Education : s.n.

**Kniberg, Henrik.** *SCRUM Y XP DESDE LAS TRINCHERAS*. . 2007.

jQuery. [En línea] <http://jquery.org/>.

LA AUTO-GESTIÓN DEL CONOCIMIENTO EN INGLÉS COMO PROCESO INTEGRADOR DE COMPETENCIAS GENÉRICAS EN LOS CENTROS DE AUTO-APRENDIAJE Y SERVICIOS DE IDIOMAS EXTRANJEROS. *UCIENCIA 2012*. [En línea] 2012. [uciencia.uci.cu/es/node/2650](http://uciencia.uci.cu/es/node/2650).

Conferencia introductoria a los sistemas de bases de datos. [En línea] Departamento central de Base de datos de la Universidad de las Ciencias Informáticas, 2009. [http://eva.uci.cu/file.php/624/2.\\_Clases/Semana\\_1/1ra\\_frecuencia/MApoyo/C1\\_Introductor ia.pdf](http://eva.uci.cu/file.php/624/2._Clases/Semana_1/1ra_frecuencia/MApoyo/C1_Introductor%20ia.pdf).

**Gracia, Joaquin.** *Patrones de diseño de software orientado a objetos*. 2005.

**Pressman, Roger S.** *Ingeniería de software. Un enfoque práctico. Capítulo 8 Modelado de Análisis*. [Documento]

Rational Rose Enterprise. [En línea] 2012. <http://www.rational.com.ar/herramientas/roseenterprise.html>.

Longman English Interactive. [En línea] Pearson Education, 2008. [Citado el: 21 de 5 de 2012.] <http://www.longmanenglishinteractive.com/>.

**BUENO, E.** *Gestión del conocimiento, aprendizaje y capital intelectual*. Madrid : Boletín del Club Intellect, 1999.