

**Universidad de las Ciencias Informáticas  
FACULTAD 1**



**Título: Implementación del Portal de Calidad de la  
Facultad 1**

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

**Autores:**

Anaisel Hernández Izquierdo  
Yanny Leyva Fajardo

**Tutor:**

Ing. Dailiany Soler Torriente

**Ciudad de la Habana, Junio 2009  
“Aniversario 50 de la Revolución.”**

*“La responsabilidad nuestra es luchar porque la calidad del producto que aquí se haga sea de las mejores y la mejor posible...”*

*Ernesto “Che” Guevara*

### DECLARACIÓN DE AUTORÍA

Declaramos que somos los únicos autores del trabajo titulado: Implementación del Portal de Calidad de la Facultad 1 y autorizamos 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 15 días del mes de junio del año 2009.

Anaisel Hernández Izquierdo

Yanny Leyva Fajardo

Ing. Dailiany Soler Torriente.

## **DATOS DE CONTACTO**

El tutor de este trabajo es la Ing. Dailiany Soler Torriente, profesor de la facultad 1. Imparte la asignatura de Matemática Discreta, Dos años de experiencia laboral.

### Agradecimientos

*Queremos Agradecer en primer lugar la oportunidad de cursar estudios en la primera Universidad surgida al calor de la batalla de ideas impulsada por nuestro Comandante en Jefe y por el privilegio de integrar el destacamento de la “Tropa Futuro”. Gracias a la Revolución por las múltiples oportunidades para superarnos.*

*A nuestros padres, por el apoyo, la confianza y el amor, por sus regaños, por ser nuestros principales guías y por ayudarnos hasta aquí.*

*Al terminar esta etapa de la vida, queremos expresar un profundo agradecimiento a todas las personas que con su ayuda, apoyo y comprensión nos alentaron para lograr este hermoso sueño que hoy se convierte en realidad.*

*A nuestros amigos, no quisiéramos mencionarlos porque, puede que se nos escape alguno, ellos saben quiénes son.*

*A mi mamá: que sin ti no hubiera podido realizar este sueño, tu amor, ayuda y dedicación han sido importante en toda mi vida universitaria, por haberme dado la vida y porque este también es tu sueño.*

*A mis hermanos y primos: para que este sueño les sirva de ejemplo.*

*A mi papá: por darme su apoyo en todo y su amor.*

*A mi padrastro: por el incondicional cariño y por quererme como una hija más.*

*A mi novio: por todo su amor y comprensión.*

*Muy especialmente a alguien que no tuvo la oportunidad de verme alcanzar tantos logros, pero estoy segura que hubiera estado muy orgullosa de mí, a mi bisabuela Mimi.*

*En fin a toda mi familia porque este sueño también es parte de ustedes.*

*A mis amigos, ¡los de verdad!: Arturo, Yen, Yaneida, Iyugnis, Yaima, Carlos, Joan Cabo Mijares, Tania y Sarais.*

*Anaisel Hernández Izquierdo.*

*Son sin duda estas líneas las más difíciles de escribir para mí, pues han sido muchas las personas que de una forma u otra me brindaron su mano en los momentos difíciles e hicieron posible que este día llegase. Quisiera agradecer ante todo a mis padres, por estar siempre cuando los necesito, por depositar toda su confianza en el empeño de hacer de mí el hombre que soy, por su infinito amor y comprensión... gracias de todo corazón... sepan que estoy orgulloso de ser su hijo. A mi hermana, por ser faro de luz e inspiración en los momentos más difíciles, por trazarme el camino a seguir y hacerme creer que solo es realmente tuyo aquello que aprendes y que eso jamás te podrá ser arrebatado.*

*A toda mi familia, por estar siempre pendientes de mí, por ser la mejor familia que se puede tener. A mis amigos, por creer en mí, por brindarme su apoyo en todo momento. A mis compañeros de estudio que han batallado junto a mí en los momentos agradables y difíciles.*

*Agradecer también a los que me cerraron las puertas y se encargaron de que no todos mis amaneceres fueran hermosos; por ustedes conocí mi voluntad de trabajo y fuerza para marcar el camino aunque se levante el polvo.*

*Yanny Leyva Fajardo.*

### Resumen

La Universidad de las Ciencias Informáticas tiene la misión de impulsar el desarrollo de la producción de software. Para lograrlo, es necesario que los productos tengan la calidad requerida.

En la facultad 1, el asesor de calidad y los aseguradores de la misma de cada proyecto no cuentan con una aplicación web para la gestión de la información de calidad de software. Por lo que el presente trabajo de diploma, tiene como objetivo principal, implementar un portal web para mejorar la gestión de la información de la calidad de los proyectos productivos de la facultad 1.

El sistema posibilitará que se logre un adecuado proceso de gestión de información relativa a la calidad mediante un portal web fácil de administrar y actualizar, así como dará la posibilidad de divulgar las principales actividades que se planifiquen.

El desarrollo del portal web estuvo guiado por las especificaciones que propone la metodología XP, obteniendo los artefactos de las diferentes fases de trabajo, como la exploración y planificación, diseño, desarrollo y pruebas.

Para la propuesta de solución se emplearon herramientas libres y de código abierto, cuya selección fue el resultado de un estudio comparativo entre las tendencias y tecnologías actuales.

En el documento se especifican de forma detallada las funcionalidades que debe brindar el sistema a sus clientes. Se propone además un diseño basado en el uso de un Sistema de Gestión de Contenido que facilite la implementación del sistema propuesto.

**PALABRAS CLAVE:** gestión, información, portal, proceso, desarrollo, aplicación web.



## Índice

Introducción 1.....	1
Capítulo 1: Fundamentación Teórica.....	6
1.1. Introducción.....	6
1.2. Aplicaciones web.....	6
1.2.1. Ventajas del uso de aplicaciones web.....	6
1.2.2. Tipos de aplicaciones web.....	7
1.3. Portales.....	8
1.3.1. Tipos de portales.....	8
1.3.2. Diferencia entre portal y sitio web.....	9
1.3.2.1. Portal.....	9
1.3.2.2. Sitio web.....	10
1.4. Análisis de soluciones existentes.....	10
1.5. Tecnologías y tendencias actuales a considerar.....	12
1.6. Sistemas de Gestión de Contenidos.....	12
1.6.1. Criterios para la selección de un CMS:.....	13
1.6.2. Tipos de CMS.....	14
1.6.3. CMS Drupal.....	15
1.7. Lenguaje de programación.....	17
1.7.1. PHP.....	17
1.8. Sistemas de Gestión de Bases de Datos.....	17
1.8.1. PostgreSQL.....	18
1.9. Servidores web.....	20
1.9.1. Apache.....	20
1.10. Metodologías de Desarrollo de Software.....	21
1.10.1. El Proceso Unificado de Modelado.....	21
1.10.2. Programación Extrema.....	22

1.10.3. Desarrollo Guiado por la Funcionalidad.....	23
1.11. Herramientas utilizadas para el desarrollo del sistema.....	23
1.11.1. Visual Paradigm.....	23
1.11.2. Zend Studio.....	24
1.12. Fundamentación de las herramientas seleccionadas.....	24
1.13. Conclusiones.....	25
Capítulo 2: Características del Sistema.....	26
2.1. Introducción.....	26
2.2. Conceptos sobre Calidad de Software.....	26
2.3. Arquitectura de la Información.....	26
2.3.1. Objetivos.....	26
2.3.2. Definición de Audiencia.....	27
2.3.3. Definición de contenidos y servicios del portal.....	28
2.3.4. Definición de la estructura del portal.....	28
2.3.4.1. Estructura del Portal de Calidad de la Facultad 1.....	29
2.3.5. Mapa de navegación del Portal de Calidad.....	29
2.3.6. Elementos de los Sistemas de Navegación.....	30
2.3.7. Diseño de la estructura de la página principal.....	31
2.4. Requerimientos funcionales del sistema.....	32
2.5. Requerimientos no funcionales del sistema.....	34
2.6. Personas relacionadas con el sistema.....	36
2.7. Aplicación de la metodología seleccionada.....	37
2.7.1. Fase de exploración.....	37
2.7.1.1. Historias de usuarios.....	37
2.7.2. Fase de planificación.....	43
2.7.2.1. Estimación de esfuerzos por historias de usuario.....	43
2.7.2.2. Plan de iteraciones.....	44

---

2.7.2.3. Plan de duración de las iteraciones.....	45
2.7.2.4. Plan de entregas .....	46
2.8. Conclusiones.....	46
Capítulo 3: Construcción del Sistema.....	48
3.1. Introducción.....	48
3.2. Diseño.....	48
3.2.1. Estructura de Drupal .....	48
3.2.1.1. Módulos de Drupal .....	50
3.2.1.2. Módulos del Portal de Calidad.....	50
3.2.2. Patrones de diseño en Drupal .....	52
3.3. Implementación .....	53
3.3.1. Iteración 1 .....	54
3.3.1.1. Tareas generadas por cada historia de usuario.....	54
3.3.2. Iteración 2 .....	61
3.3.2.1. Tareas generadas por cada historia de usuario.....	62
3.3.3. Iteración 3 .....	66
3.3.3.1. Tareas generadas por cada historia de usuario.....	66
3.3.4. Desarrollo de los módulos en Drupal.....	68
3.3.5. Diagrama de despliegue.....	70
3.4. Pruebas.....	71
3.4.1. Pruebas unitarias .....	71
3.4.2. Pruebas de aceptación.....	72
3.5. Conclusiones.....	72
Conclusiones.....	73
Recomendaciones.....	74
Trabajos citados .....	75
Bibliografía.....	77
Glosario de Términos .....	79

Anexos.....82

**Índice de tablas**

Tabla 2. 2 Personas relacionadas con el sistema .....36

Tabla 2. 3 HU Autenticar.....37

Tabla 2. 4 HU Gestionar documentación.....38

Tabla 2. 5 HU Gestionar información de proyectos .....38

Tabla 2. 6 HU Gestionar listado de miembros por proyectos .....38

Tabla 2. 7 HU Gestionar noticias .....38

Tabla 2. 8 HU Gestionar Fórum .....39

Tabla 2. 9 HU Gestionar Encuesta.....39

Tabla 2. 10 HU Gestionar Enlaces.....39

Tabla 2. 11 HU Buscar Información .....39

Tabla 2. 12 HU Imprimir documentos.....40

Tabla 2. 13 HU Autenticar contra dominio LDAP.....40

Tabla 2. 14 HU Gestionar cronograma de revisiones.....40

Tabla 2. 15 HU Gestionar preguntas frecuentes .....40

Tabla 2. 16 HU Mostrar el mapa de navegación .....41

Tabla 2. 17 HU Gestionar solicitud de revisión.....41

Tabla 2. 18 HU Insertar resultado de revisión .....41

Tabla 2. 19 HU Mostrar reporte de resultado de revisión. ....41

Tabla 2. 20 HU Insertar resultado de auditoría.....42

Tabla 2. 21 HU Mostrar reporte de resultado de auditoría.....42

Tabla 2. 22 HU Gestionar eventos .....42

Tabla 2. 23 HU Mostrar cantidad de no conformidades a los jefes de proyectos .....42

Tabla 2. 24 HU Publicar cursos de formación. ....43

Tabla 2. 25 Estimación de esfuerzos por HU .....43

Tabla 2. 26 Plan de duración de las iteraciones .....45

Tabla 2. 27 Plan de entrega.....46

Tabla 3. 1 Historias abordadas en la primera iteración.....54

Tabla 3. 2 Tarea #1 HU Autenticar.....54

Tabla 3. 3 Tarea #1 HU Gestionar documentación. ....55

Tabla 3. 4 Tarea #2 HU Gestionar documentación. ....55

Tabla 3. 5 Tarea #1 HU Autenticación contra dominio LDAP. ....55

Tabla 3. 6 Tarea #1 HU Gestionar Información de proyecto.....56

Tabla 3. 7 Tarea #2 HU Gestionar Información de proyecto.....56

Tabla 3. 8 Tarea #3 HU Gestionar Información de proyecto.....56

Tabla 3. 9 Tarea #4 HU Gestionar Información de proyecto.....57

Tabla 3. 10 Tarea #1 HU Gestionar cronograma de revisiones.....57

Tabla 3. 11 Tarea #2 HU Gestionar cronograma de revisiones.....57

Tabla 3. 12 Tarea #3 HU Gestionar cronograma de revisiones.....58

Tabla 3. 13 Tarea #4 HU Gestionar cronograma de revisiones.....58

Tabla 3. 14 Tarea #1 HU Gestionar solicitud de revisión.....58

Tabla 3. 15 Tarea #2 HU Gestionar solicitud de revisión.....59

Tabla 3. 16 Tarea #3 HU Gestionar solicitud de revisión.....59

Tabla 3. 17 Tarea #4 HU Gestionar solicitud de revisión.....59

Tabla 3. 18 Tarea #1 HU Insertar resultado de revisión .....60

Tabla 3. 19 Tarea #1 HU Mostrar reporte de resultado de revisión.....	60
Tabla 3. 20 Tarea #1 HU Insertar resultado de auditoría.....	60
Tabla 3. 21 Tarea #1 HU Mostrar resultado de auditoría.....	61
Tabla 3. 22 Historias abordadas en la segunda iteración.....	61
Tabla 3. 23 Tarea #1 HU Gestionar miembros por proyecto.....	62
Tabla 3. 24 Tarea #2 HU Gestionar miembros por proyecto.....	62
Tabla 3. 25 Tarea #3 HU Gestionar miembros por proyecto.....	62
Tabla 3. 26 Tarea #4 HU Gestionar miembros por proyecto.....	63
Tabla 3. 27 Tarea #1 HU Gestionar noticias.....	63
Tabla 3. 28 Tarea #2 HU Gestionar noticias.....	63
Tabla 3. 29 Tarea #3 HU Gestionar noticias.....	64
Tabla 3. 30 Tarea #4 HU Gestionar noticias.....	64
Tabla 3.31 Tarea #1 HU Gestionar Fórum.....	64
Tabla 3.32 Tarea #1 HU Gestionar encuesta.....	65
Tabla 3. 33 Tarea #1 HU Gestionar enlaces.....	65
Tabla 3. 34 Tarea #1 HU Publicar cursos de formación.....	65
Tabla 3.35 Tarea #2 HU Publicar cursos de formación.....	66
Tabla 3. 36 Historias abordadas en la tercera iteración.....	66
Tabla 3. 37 Tarea #1 HU Buscar información.....	66
Tabla 3. 38 Tarea #1 HU Imprimir información.....	67
Tabla 3. 39 Tarea #1 HU Gestionar preguntas frecuentes.....	67
Tabla 3.40 Tarea #1 HU Mostrar el mapa de navegación.....	67
Tabla 3. 41 Tarea #1 HU Gestionar eventos.....	68
Tabla 3. 42 Tarea #1 HU Mostrar cantidad de no conformidades a jefes de proyectos.....	68

## Índice de Figuras

Figura 1 Mapa de Navegación del Portal de Calidad. ....	30
Figura 2 Diseño de la estructura de la página principal.....	31
Figura 3 Diagrama de paquete de Drupal. ....	49
Figura 4 Diagrama de despliegue. ....	71

# INTRODUCCIÓN

En el mundo de hoy, ha tomado gran auge la Informática y como parte de ella la producción de software. Es por ello, que una de las principales tareas del Gobierno Cubano, es desarrollar la Industria del Software, no solamente por los beneficios que se aportan al desarrollar sistemas informáticos para uso interno, sino también con el fin de insertarse en el mercado de software a nivel mundial, ya que este tiene grandes perspectivas económicas.

Pero en este camino, no todo es fácil ni se obtienen buenos resultados, pues surgen algunos inconvenientes en el desarrollo: grandes retrasos en la programación, inconsistencia en su funcionamiento, etc.; pero lo más importante es la falta de calidad, punto de gran interés e importancia para el logro de eficiencia y aceptación de los sistemas.

Es claro que si un sistema presenta errores al ser utilizado, ese producto pierde confiabilidad a los ojos del usuario que podría desecharlo por ser un producto defectuoso. La Universidad de las Ciencias Informáticas (UCI), como centro que se dedica al desarrollo de programas informáticos, se ve obligada a velar celosamente por la calidad de los productos que se desarrollan, para lograr que los mismos sean competitivos y paralelamente a ello, minimizar en lo posible los inconvenientes mencionados anteriormente.

Hoy en día, la calidad de software tiene mucha importancia. Para garantizar que los proyectos obtengan, como resultado de un proceso de desarrollo de software, productos de altísima calidad, la UCI se trazó como estrategia la formación de un Grupo de Aseguradores de Calidad en cada proyecto de cada Facultad, guiados por los Asesores de Calidad de las mismas.

Como el proceso de gestión de la calidad en la UCI es incipiente, es necesario que se conozca sobre el tema. Además que los líderes de proyecto, y otras personas relacionadas con el proceso de calidad, accedan o tengan la documentación necesaria para solicitar la revisión de un producto. Posibilitar que los mismos se preparen y posean conciencia de la importancia de llevar a cabo el proceso. Por eso es necesario que todos los que participan conozcan bien los documentos que deben ser revisados y confeccionados.



Actualmente el Grupo de Aseguradores de Calidad de la facultad 1 no cuenta con un sitio oficial para la gestión de la información relativa de los procesos de la calidad de los proyectos que ejecuta, a través del cual se pueda gestionar y consultar información referente a su estructura productiva, las normas de calidad, temas relacionados a planes y resultados relevantes en las auditorías y las revisiones, mantener disponible y actualizado el expediente de cada uno de los proyectos.

Por lo anterior, el control de la información se lleva a cabo de manera manual, lo que incide de forma negativa en la eficiencia del proceso de gestión de la calidad y contribuye en muchos casos al deterioro o pérdida de la información referente a dicha actividad.

No existe manera de publicar materiales, artículos, experiencias prácticas e investigaciones propias del Grupo de Aseguradores de Calidad de la facultad 1.

Todo lo anterior expuesto provoca retrasos en la obtención de consolidados de información debido a que se emplea más tiempo del requerido en la búsqueda de soluciones óptimas. Una vez analizada la problemática expuesta surge el siguiente **problema a resolver**: ¿Cómo mejorar la eficiencia de los procesos de gestión de la información de la calidad en los proyectos productivos de la facultad 1?

El **objeto de estudio** de este trabajo se centra en el proceso de gestión de la información de la calidad para los proyectos productivos de la facultad 1 y el **campo de acción** se enmarca en la planificación y publicación de las actividades referentes a la gestión de la información de la calidad relacionada con los proyectos productivos de la facultad 1.

Como **objetivo general** se propone implementar un portal web que permita hacer más eficiente la gestión de la información de los procesos de la calidad en los proyectos productivos de la facultad 1.

Para lograr el objetivo propuesto se proponen los siguientes **objetivos específicos**:

- Establecer el marco conceptual sobre los sistemas de gestión de contenido para la web.
- Determinar las necesidades informativas de la facultad relativa a sus procesos de calidad del software.

- Diseñar la arquitectura de información del portal web.
- Diseñar los servicios y/o funcionalidades del portal web.
- Implementar la solución en un sistema de gestión de contenidos.
- Garantizar la calidad y el funcionamiento de la propuesta de solución.

Como **idea a defender** contar con un portal web para la gestión de la información de la calidad, facilitará que los proyectos productivos de la facultad 1 tengan conocimiento sobre la planificación y publicación de las actividades.

Para dar cumplimiento al objetivo trazado se proponen las siguientes **tareas de la investigación**:

- Revisar bibliografía existente sobre el tema tanto a nivel internacional, nacional como en la UCI.
- Realizar un estudio de aplicaciones o propuestas sobre esta área, utilizadas para comprobar la satisfacción del cliente.
- Establecer la situación actual de las herramientas de gestión de contenido.
- Obtener los requisitos funcionales del Portal de Calidad.
- Diseñar la arquitectura de información.
- Establecer pautas para el diseño de la interfaz.
- Seleccionar las herramientas de implementación.
- Implementar el Portal de Calidad.

Para darle cumplimiento a las tareas propuestas con anterioridad, se utilizarán los **métodos científicos** siguientes:

### **Métodos Teóricos**

**Analítico – sintético:** Facilita el entendimiento del fenómeno en el que se trabaja, es más útil la división de este en diferentes fases, y de esta forma descubrir sus características generales, lo que ayuda a seguir una correcta investigación.

Estudiando y analizando todo lo relacionado al tema específico a abordar se pudo hacer una profundización de todas sus características, dígase objetivos, logros, o deficiencias que se derivan, así como también las mejoras que se pueden proponer para obtener resultados eficientes.

**Análisis histórico lógico:** Permite estudiar de forma analítica la trayectoria histórica real de los fenómenos, su evolución y desarrollo.

Se utiliza este método porque uno de los objetivos planteados fue el de analizar el marco conceptual sobre los sistemas de gestión de contenido para la web, conociendo de esta forma la evolución que ha tenido el desarrollo de portales web, se emplea durante el proceso de diseño y mejoramiento de las funcionalidades que satisfacen o podrían satisfacer las demandas de gestión de calidad.

### **Métodos Empíricos**

**Entrevista:** Es una conversación planificada entre el investigador y el entrevistado para obtener información. Su uso constituye un medio para el conocimiento cualitativo de los fenómenos.

Se realizan entrevistas al cliente: Eileen Llano Castro con el objetivo de obtener la mayor información posible sobre las necesidades del Grupo de Aseguradores de Calidad.

**Experimento:** El experimento es el método empírico para el estudio de un objeto en el cual el investigador crea las condiciones o adapta las existentes para el esclarecimiento de las propiedades, leyes y relaciones del objeto, para verificar una hipótesis, una teoría o un modelo. El experimento puede ser transformador o comprobador.

Se utiliza este método ya que uno de los objetivos propuestos fue el de garantizar la calidad y el funcionamiento de la propuesta de solución, por lo que es necesario la realización de pruebas para dar cumplimiento a dicho objetivo.

Se espera como **aporte práctico** brindarle al personal encargado del proceso de gestión de la información de la calidad y proyectos productivos de la facultad 1, una aplicación web con las funcionalidades requeridas. También se provee al mismo de una interfaz administrativa para el manejo de los contenidos facilitando la edición y actualización de estos, sin la necesidad por parte del personal, de conocer los elementos técnicos para su uso. Además se pretende hacerle llegar al personal que utilice el portal la

información más actualizada y precisa buscando la conformidad del cliente y la necesidad del mismo de visitarlo con frecuencia.

El documento está estructurado en tres capítulos, en los cuales se describe todo el proceso por el cual se transita durante el desarrollo del trabajo:

**Capítulo 1. Fundamentación teórica:** Este capítulo comprende el estado del arte sobre el tema tratado, las tendencias actuales que existen sobre los portales de gestión de información sobre calidad nacional e internacional. Se describe la justificación de las tecnologías, metodología y herramientas utilizadas para la solución del problema.

**Capítulo 2. Características del Sistema:** Este capítulo plantea los requisitos funcionales y no funcionales que debe tener el sistema para lograr óptimos resultados, además de hacer referencia a las personas vinculadas a este y detalla los artefactos generados en la fase de Exploración y Planificación, teniendo especial atención en las Historias de Usuario.

**Capítulo 3. Construcción del Sistema:** Este capítulo presenta el diseño, implementación y las pruebas que complementan el portal.

# CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA

## 1.1. Introducción

En el presente capítulo se muestra un estudio sobre las definiciones, ventajas y tipos de aplicaciones web. Se aborda, además, el estado del arte de los portales de Gestión de Calidad existentes. Por último se realiza un estudio de las tendencias, tecnologías y metodologías más usadas en la actualidad y se concluye con la selección de las herramientas a utilizar durante el desarrollo de la aplicación.

## 1.2. Aplicaciones web

Antes de entrar en los detalles propios de la aplicación a desarrollar, cabe destacar, que para cumplir los objetivos de este trabajo, se debe desarrollar un sistema que les posibilite a múltiples usuarios tener acceso al mismo tiempo a la aplicación, posibilitando la obtención rápida de la información, sin necesidad de compartir escritorios, ni del intercambio físico directo entre los aseguradores de calidad de cada proyecto en la facultad y sus usuarios, lo que no es posible mediante aplicaciones de escritorio. En este caso, la opción óptima es el uso de una aplicación web.

Una aplicación web es un sistema informático que los usuarios utilizan accediendo a un servidor web a través de Internet o de una intranet. Estas son populares debido a la practicidad del navegador web como cliente ligero. Entre estas se pueden encontrar los webmails, wikis, weblogs, tiendas en línea y la Wikipedia.

Dentro de las aplicaciones web, se encuentran los servicios web, estos actualmente son bastante sencillos. Proporcionan un lenguaje y una sintaxis independiente de la plataforma para intercambiar datos complejos mediante mensajes.

### 1.2.1. Ventajas del uso de aplicaciones web

Además de la ventaja de posibilitar el acceso a varios usuarios simultáneamente al sistema, estas poseen un gran número de ventajas, entre las que se encuentran las que se muestran a continuación:

- Las aplicaciones web, son actualizadas con el último lanzamiento sin requerir que el usuario tome acciones pro-activas, y sin necesitar llamar la atención del usuario o interferir con sus hábitos de trabajo con la esperanza de que va a iniciar nuevas descargas y procedimientos de instalación.

- Tienen un precio menor que las aplicaciones de escritorio, estas no requieren la infraestructura de distribución, soporte técnico y marketing requerido por las aplicaciones descargables de software tradicionales. Esto permite que las aplicaciones online cuesten una fracción de sus contrapartes descargables.
- Los datos están online, es decir, no es necesario encontrarse precisamente en la computadora en la que se instaló el software para tener acceso al mismo, sin necesidad de compartir escritorios.
- En el caso de las personas o instituciones que usan las aplicaciones web, encuentran en las mismas un medio de promoción rápido y eficiente.
- Cuenta con una mayor inmediatez de acceso, estas no necesitan ser descargadas, instaladas y configuradas. (16)

### 1.2.2. Tipos de aplicaciones web

Las aplicaciones web se dividen en tres grupos fundamentales: Informativas, De negocio y Portales.

- Las aplicaciones web informativas, son las más sencillas de realizar. En las que sólo se muestra información de un tema determinado, donde la página principal es la de mayor importancia a la hora de su elaboración, ya que es la que da la idea del contenido que se encontrará en la aplicación.
- Las aplicaciones web de negocio, incluyen las aplicaciones informativas y opciones donde el cliente selecciona lo que necesita de un producto o servicio, en esta el intercambio de información es más complejo, en su forma más sencilla, luego de la selección de los elementos que el usuario necesita, el sistema le pide los datos y los almacena en la base de datos con toda la información ingresada por el usuario.
- Los portales son el grupo más complejo de las aplicaciones web. Un portal de Internet es conocido como “Puerta o puerto”, es decir, es un sitio web que sirve como punto de partida para la navegación en la red. Posibilitando de forma fácil y sencilla, el acceso a una serie de recursos y de servicios, entre los que suelen encontrarse facilidades para la localización de sitios como son los buscadores, noticias, buzones de correo, directorios telefónicos y foros de discusión con temas de interés general.

### 1.3. Portales

Un portal es un conjunto de páginas web o sitio web que sirve como un punto de entrada único a la información de una institución o empresa. Los portales deben estar organizados de forma tal, que para los usuarios sea fácil, cómodo e intuitivo encontrar la información que desean. Por este motivo, es bastante común que los portales cuenten con alguna especie de índice o mecanismo de búsqueda.

Pudiera considerarse también que un portal es un punto de entrada a Internet donde se organizan los contenidos y concentran los servicios y productos para ayudar a los usuarios a realizar cuanto necesiten, o al menos para que pueda encontrar allí todo cuanto utiliza cotidianamente, sin necesidad de salir de dicho sitio.

Un portal es la vía para lograr el acceso al recurso necesario, él facilita la comunicación y el intercambio de información entre los usuarios de una organización y esto puede afirmarse debido a que el elemento que lo distingue es, que este constituye una red y que a partir de su página de presentación, los usuarios pueden acceder a las aplicaciones, productos, servicios, documentos, otros sitios, aplicaciones propias o compartidas que necesitan. (1)

#### 1.3.1. Tipos de portales

Pueden existir diferentes tipos de portales en función de sus usuarios, éstos son:

- **Portal Horizontal:** Su objetivo son los usuarios en general. Suelen ofrecer motores de búsqueda, noticias, e-mail y otras posibilidades de comunicación. Los contenidos son absolutamente críticos, y se está evolucionando hacia la propia personalización del portal.
- **Portal Vertical:** Son portales especializados en determinados temas, que buscan público y tienen un objetivo específico. Se pueden a su vez clasificar en función de sus objetivos:
  - **Portal Intranet:** Comunicación corporativa para los empleados.
  - **Portal Extranet:** Comunicación corporativa para los proveedores.
  - **Portal Vertical:** Comunicación corporativa con clientes.

La diferencia radical entre los portales horizontales y verticales, es que los portales horizontales ofrecen sus servicios a los usuarios en general, sin importar cuál será el contenido del portal, mientras que los

portales verticales enfocan sus servicios para alguna institución determinada mediante la implementación de Intranets, Extranets o Portales Verticales.

El tipo de portal a implementar es un portal vertical ya que se analizará, diseñará e implementará el mismo para un tema en específico, Calidad de Software.

### **1.3.2. Diferencia entre portal y sitio web**

#### **1.3.2.1. Portal**

Portal es un término, sinónimo de puente, para referirse a un sitio web que sirve o pretende servir como un sitio principal de partida para las personas que se conectan al World Wide Web. Un sitio web no alcanza el rango de portal sólo por tratarse de un sitio robusto o por contener información relevante. Un portal es más bien una plataforma de despegue para la navegación en la web.

Los portales tienen gran reconocimiento en Internet por el poder de influencia que poseen sobre grandes comunidades. Se entiende por portal algo más genérico, con mayor ámbito de actuación y poder de información. Es la entrada a un mercado de información dónde vas a consumir información y encuentras multitud de puestos (webs, buscador, enlaces, utilidades, herramientas, programas, documentación, etc.).

Un sitio web sería una dirección que contiene información más específica y relacionada con un sólo tema, campo u organización (empresa, institución, etc.). Aunque te suministre enlaces, siempre será mucho más limitada que un portal.

Desde el punto de vista funcional, un portal se diferencia de un sitio web convencional debido a que incluye muchas más secciones, como pueden ser noticias, eventos, directorios, que requieren estar siempre actualizados. Por otro lado, normalmente se caracterizan por administrar claves de acceso, destinadas a restringir la visita a determinadas páginas con información de alto valor. El portal es considerado un intermediario de información que tiene como fuente de ingreso la publicidad de empresas que ahí se anuncian. Realmente el portal es la evolución de un sitio web, más en el ámbito conceptual que en el tecnológico, se ha ido evolucionando en contenidos y servicios.



### 1.3.2.2. Sitio web

Es un conjunto de archivos electrónicos y páginas web referentes a un tema en particular, que incluye una página inicial de bienvenida, generalmente denominada home page, con un nombre de dominio y dirección en Internet específicos. Empleados por las instituciones públicas y privadas, organizaciones e individuos para comunicarse con el mundo entero.

En el caso particular de las empresas, este mensaje tiene que ver con la oferta de sus bienes y servicios a través de Internet, y en general para hacer eficiente sus funciones de mercadotecnia. A veces se utiliza erróneamente el término página web para referirse a un sitio web. Una página web es parte de un sitio web y es un único archivo con un nombre de archivo asignado, mientras que un sitio web es un conjunto de archivos llamados páginas web. Si se compararan con un libro, un sitio web sería el libro entero y una página web de ese sitio web sería un capítulo de ese libro.

El único requisito es que el equipo en el que residan los documentos esté conectado a la red mundial de Internet. Este equipo de cómputo o servidor web, como se le denomina técnicamente, puede contener más de un sitio web y atender concurrentemente a los visitantes de cada uno de los diferentes sitios. Los sitios web pueden ser de diversos géneros, destacando los sitios de negocios, servicio, comercio electrónico en línea, imagen corporativa, entretenimiento y sitios informativos.

### 1.4. Análisis de soluciones existentes

La gran red mundial de ordenadores, Internet, se caracteriza por la enorme cantidad de información que contiene y que en la mayoría de los casos es accesible de forma libre y gratuita. La misma se ha convertido en una forma fácil, atractiva y universal en la utilización de la red para encontrar múltiples sitios web de diversos temas, los dedicados a la calidad del software constituye un tema interesante dentro de la rica gama de información disponible que se visualiza a través de Portales de Información. Ejemplo de portales de calidad de software:

**Exact Software.com y Calidaddelsoftware.com:** (español). Sus principales funciones son: la publicación de libros, documentos o artículos referentes a estándares, métricas, normas, eventos a desarrollarse o noticias. También muestran definiciones y conceptos que son de gran utilidad para entender todo lo que

lleva en sí, un proceso de calidad. Publican modelos a seguir para obtener productos con calidad, realizan consultas y fórums donde se puede debatir cualquier duda que tenga el usuario.

**Inteco.es:** Gestiona información sobre calidad del software, proporciona asesoramiento y formación mediante cursos, seminarios y encuentros a empresas sobre calidad del software.

**Montouso.com:** (Software Engineering Inc.) Ofrece seminarios, cursos y fórums sobre calidad del software. Publica artículos y además realiza pruebas de revisión a productos. Panamá.

**Ati.es:** (Asociación de Técnicos de Informática) El Grupo de Calidad del Software tiene como objetivo la difusión y el debate sobre todos los aspectos relacionados con la calidad del software y de los sistemas, así como la promoción de buenas prácticas de calidad y metodologías entre toda la comunidad profesional y universitaria. Mediante cursos, seminarios, fórums y publicaciones de artículos y libros. España.

**E-Quality.net:** Consorcio de empresas especializadas en prueba de software. Realización de pruebas de software, calificación de productos de software (sello de calidad), y la capacitación de pruebas al personal de desarrollo del producto. Publica libros y artículos con estándares y métricas. México.

**Calidadsoft.prod.uci.cu:** Sus principales funciones son: Oferta de servicios de formación, servicios de diagnóstico de calidad en procesos de desarrollo de software, servicio de certificación en el área de mejoramiento continuo de la calidad. Contribuir a la identificación, generación, promoción y adopción de estándares, normas y mejores prácticas relacionadas con la calidad en la Ingeniería de Software. Desarrollar acuerdos con todo tipo de instituciones nacionales e internacionales con el objeto de realzar las actividades, proyectos y programas inherentes al cumplimiento de su misión. Publica libros artículos sobre calidad del software, además posee fórums para debate.

**Portal de Calidad Facultad 2:** Consulta online de documentación actualizada sobre el tema de la calidad de software. Publica informaciones y/o noticias sobre trabajo actual y temas de interés para el proyecto, plantillas y documentación sobre trabajos anteriores, así como record de los proyectos revisados. Gestiona ubicación de puestos de trabajo del proyecto, etc.

Durante esta investigación sobre el proceso de gestión de la información se destaca la utilización de tecnologías web, centrándose en los portales web para llevar a cabo este proceso, sacando como premisa

que la realización de un portal web se convertiría en la solución adecuada para dar cumplimiento a las necesidades del Grupo de Aseguradores de la Calidad de cada proyecto de la facultad 1, de gestionar la información referente a la calidad de software y de los proyectos que estén en desarrollo en la misma.

### 1.5. Tecnologías y tendencias actuales a considerar

Las aplicaciones web están actualmente entre las soluciones más usadas y populares para automatizar procesos de gestión de información. En la UCI las tendencias a la hora de desarrollar aplicaciones web son: la utilización de Software Libre y CMS. El software libre porque ofrece un gran número de ventajas como por ejemplo, se le pueden realizar las modificaciones que se crean necesarias para adaptarlo a características específicas que pueda tener el producto a desarrollar, existen además numerosas comunidades de programadores en cuyas páginas web se pueden encontrar códigos predefinidos y reutilizables que resuelvan muchos de los problemas a los que se intenta dar solución como parte del proyecto, entre otras muchas ventajas.

En el caso de los CMS una de las principales razones de su creciente popularidad es que estos surgen precisamente como una forma de hacer más fácil el desarrollo de aplicaciones web que manejen grandes cantidades de información, que requieran de actualizaciones constantes o para las cuales se prevea un crecimiento futuro. Además los CMS permiten manejar de manera independiente el contenido y el diseño y dan la posibilidad de una fácil y controlada publicación en el sitio a varios editores.

### 1.6. Sistemas de Gestión de Contenidos

Dada la necesidad de manipulación de grandes cantidades de información en el Portal de Calidad de la facultad 1, se llegó a la conclusión que la vía óptima de desarrollar la aplicación es mediante el uso de un Sistema de Gestión de Contenidos (CMS). Los Sistemas de Gestión de Contenidos se han convertido en una tendencia general adoptada por los desarrolladores de aplicaciones que manipulan contenido web. (6) Estos sistemas de gestión de contenidos presentan ventajas que se explican a continuación:

- Generan, gestionan y publican la información con la máxima eficiencia y productividad, ya que son herramientas que permiten al responsable de contenido de un sitio web o portal publicar, editar, borrar, actualizar y administrar la información del sitio sin conocimiento alguno de HTML. Un CMS

no debe presentar ninguna dificultad técnica para el responsable de la empresa encargado de su gestión.

- Basado en un entorno 100% web ya que la finalidad de un CMS es la creación de aplicaciones y sitios web dinámicos de forma rápida y eficiente. Permite la integración de contenidos de múltiples fuentes y formatos (textos, imágenes, audio, video).
- Sencilla administración de la web o portal pues la curva de aprendizaje es extremadamente baja. La información se introduce rellenando unos sencillos formularios a través de un navegador web. La administración completa de la web o portal se realiza desde una única aplicación web.
- Ideales para pequeñas y medianas empresas pues los CMS son sistemas flexibles y modulares, que incorporan la última tecnología Internet y que permite a cualquier persona generar, gestionar y publicar contenidos de cualquier tipo: texto, imágenes, audio, vídeo.
- Solución adaptable mediante módulos: Las ventajas de un CMS son claras y evidentes. Los módulos independientes y los numerosos componentes pueden combinarse para obtener la solución específica para unas necesidades concretas. Facilita la gestión autónoma de un sitio sin necesidad de técnicos informáticos ni mantenimiento. De esta manera favorece la participación de distintos colaboradores debido a la facilidad de uso y a una concepción transparente de su funcionamiento.

### 1.6.1. Criterios para la selección de un CMS:

- Editor de texto WYSIWYG.
- Herramienta de búsqueda.
- Comunicación entre los usuarios (fóruns, correo electrónico, chat).
- Noticias y Artículos.
- Ciclo de trabajo (workflow) con diferentes perfiles de usuarios y grupos de trabajo.
- Fechas de publicación y caducidad.

- Webs personales.
- Carga y descarga de documentos y material multimedia.
- Envío de páginas por correo electrónico.
- Páginas en versión imprimible.
- Personalización según el usuario.
- Disponibilidad o posibilidad de traducción al catalán y al castellano.
- Soporte de múltiples formatos (HTML, Word, Excel, Acrobat, etc.).
- Soporte de múltiples navegadores (Internet Explorer, Netscape, etc.).
- Soporte de sindicación (RSS, NewsML, etc.).
- Estadísticas de uso e informes.
- Control de páginas caducadas y enlaces rotos.
- Avisos de actualización de páginas o mensajes en los fórums, y envío automático de avisos por correo electrónico.

### 1.6.2. Tipos de CMS

Existe una gran variedad de CMS, los cuales se pueden agrupar según el tipo de sitio que permiten gestionar, entre los fundamentales se encuentran:

- **Genéricos:** Ofrecen la plataforma necesaria para desarrollar e implementar aplicaciones que den solución a necesidades específicas. Pueden servir para construir soluciones de gestión de contenidos, para soluciones de comercio electrónico, blogs, portales.
- **Específicos para Organizaciones No Gubernamentales (ONG):** Nacen para cubrir las necesidades de las ONG, ofreciendo una plataforma de servicios de Internet que en ocasiones incluye además del CMS herramientas para la recaudación de fondos, etc.
- **Foros:** Sitio que permite la discusión en línea donde los usuarios pueden reunirse y discutir temas en los que están interesados. Ejemplos: phpBB, MyBB.

- **Blogs:** Publicación de noticias o artículos en orden cronológico con espacio para comentarios y discusión. Ejemplos: Wordpress, Typo.
- **Wikis:** Sitio web dónde todos los usuarios pueden colaborar en los artículos, aportando información o reescribiéndola. También permite espacio para discusiones. Indicado para material que irá evolucionando con el tiempo. Ejemplos: Mediawiki, Tikiwiki.
- **e-Commerce:** Son sitios web para comercio electrónico.
- **e-Learning:** Sirve para la enseñanza. Los usuarios son los profesores y estudiantes, se tienen aulas virtuales donde se ponen a disposición el material del curso. La publicación de un contenido por un profesor es la puesta a disposición de los estudiantes, en un aula virtual, de ese contenido. Ejemplo: Moodle.
- **Publicaciones digitales:** Son plataformas especialmente diseñadas teniendo en cuenta las necesidades de las publicaciones digitales, tales como periódicos, revistas, etc. Ejemplo: ePrints.
- **Portal:** Sitio web con contenido y funcionalidad diversa que sirve como fuente de información o como soporte a una comunidad. Ejemplos: PHPNuke, Postnuke, Drupal, Plone.

### 1.6.3. CMS Drupal

Es un sistema de gestión de contenido (CMS) que permite, tanto a individuos como a comunidades de usuarios, la publicación de contenidos en Internet, de manera sencilla, y sin tener conocimientos de lenguaje de marcas. Desarrollado dentro de la filosofía del Software Libre y de código fuente abierto bajo licencia GPL, Drupal es un sistema dinámico, puesto que el contenido de las páginas y otras configuraciones se pueden editar online mediante las herramientas que él mismo facilita. Desarrollado en PHP y mantenido por una amplia comunidad permite múltiples usuarios y distintos lenguajes de marcas. Destaca la calidad de su código, el respeto por los estándares web, un énfasis especial en la usabilidad y consistencia de todo el sistema. Su flexibilidad y adaptabilidad, así como su estructura modular, existe gran cantidad de módulos adicionales disponibles, hacen que sea adecuado para el desarrollo de sitios web con funcionalidades distintas. Algunos de los rasgos distintivos de Drupal son la organización y clasificación de contenidos mediante “taxonomías”, las URLs amigables o URLs friendlies y las características de accesibilidad, que en su caso, depende de los temas. (7)

### ¿Por qué es un sistema dinámico?

El CMS Drupal en lugar de almacenar sus contenidos en archivos estáticos en el sistema de ficheros del servidor de forma fija, el contenido textual de las páginas y otras configuraciones son almacenados en una base de datos y se editan utilizando un entorno web incluido en el producto.

Entre sus características se encuentran:

- **Ayuda on-line:** Un robusto sistema de ayuda online y páginas de ayuda para los módulos del 'núcleo', tanto para usuarios como para administradores.
- **Búsqueda:** Todo el contenido en Drupal es totalmente indexado en tiempo real y se puede consultar en cualquier momento.
- **Código abierto:** El código fuente de Drupal está libremente disponible bajo los términos de la licencia GNU/GPL. Al contrario de otros sistemas de 'blogs' o de gestión de contenido propietarios, es posible extender o adaptar Drupal según las necesidades.
- **Módulos:** La comunidad de Drupal ha contribuido muchos módulos que proporcionan funcionalidades como 'página de categorías', autenticación mediante jabber, mensajes privados, bookmarks, etc.
- **Personalización:** Un robusto entorno de personalización está implementado en el núcleo de Drupal. Tanto el contenido como la presentación pueden ser individualizados de acuerdo las preferencias definidas por el usuario.
- **URLs amigables:** Drupal usa el mod\_rewrite de Apache para crear URLs que son manejables por los usuarios y los motores de búsqueda.
- **Posee interfaz** de administración integrada en la propia web, a diferencia de otros CMS, además es multiplataforma. El sistema de temas de Drupal permite generar plantillas completamente basadas en CSS y XHTML y lo mejor de todo es que los módulos normalmente generan código válido.
- **Independencia de la base de datos:** Aunque la mayor parte de las instalaciones de Drupal utilizan MySQL, existen otras opciones. Drupal incorpora una capa de abstracción de base de datos que actualmente está implementada y mantenida para MySQL y PostgreSQL, aunque permite incorporar fácilmente soporte para otras bases de datos.

### 1.7. Lenguaje de programación

Los lenguajes de programación son herramientas que permiten crear programas y software, facilitan la tarea de programación, ya que disponen de formas adecuadas que permiten ser leídas y escritas por personas, a su vez resultan independientes del modelo de computador a utilizar.

Debido a la selección del CMS Drupal para lograr el objetivo de la investigación surge la necesidad de utilizar PHP como lenguaje de programación ya que es un lenguaje implícito en el funcionamiento del CMS

#### 1.7.1. PHP

PHP (acrónimo de PHP: Hypertext Preprocessor), es un lenguaje interpretado de alto nivel embebido en páginas HTML y ejecutado en el servidor. La meta del lenguaje es permitir rápidamente a los desarrolladores la generación dinámica de páginas. Con PHP se puede hacer cualquier cosa que se pueda realizar con un script CGI, como el procesamiento de información en formularios, foros de discusión, manipulación de cookies y páginas dinámicas. También ofrece la integración con varias bibliotecas externas, que permiten que el desarrollador haga casi cualquier cosa, desde generar documentos en PDF hasta analizar código XML. (4)

PHP ofrece una solución simple y universal para las paginaciones dinámicas de la web de fácil programación. Su diseño elegante lo hace perceptiblemente más fácil de mantener y ponerse al día en comparación con el código de otros lenguajes. Debido a su amplia distribución PHP está perfectamente soportado por una gran comunidad de desarrolladores. Como producto de código abierto, PHP goza de la ayuda de un gran grupo de programadores, permitiendo que los fallos de funcionamiento se encuentren y se reparan rápidamente. El código se pone al día continuamente con mejoras y extensiones de lenguaje para ampliar las capacidades de PHP. Este es el lenguaje de programación en el que está creado Drupal, por lo cual necesita ser instalado para que pueda ser interpretado por el servidor.

### 1.8. Sistemas de Gestión de Bases de Datos

Un Sistema Gestor o Manejador de Bases de Datos (SGBD) es un conjunto de programas que permite a los usuarios crear y mantener una Base de Datos (BD), por lo tanto, el SGBD es un software de propósito



general que facilita el proceso de definir, construir y manipular la BD para diversas aplicaciones. Pueden ser de propósito general o específico.

Se define una base de datos como una serie de datos organizados y relacionados entre sí, los cuales son recolectados y explotados por los sistemas de información de una empresa o negocio en particular.” (19)

Los sistemas de bases de datos están diseñados para gestionar grandes volúmenes de información. Generalmente, las bases de datos requieren gran cantidad de espacio de almacenamiento, por lo que las bases de datos de las organizaciones se miden en términos de gigabytes o terabytes de datos. Un gigabyte equivale a 1000 megabytes (un billón de bytes), y un terabyte equivale a un millón de megabytes (un trillón de bytes). Un sistema de bases de datos tiene como objetivo simplificar y facilitar el acceso a los datos y hacer que los tiempos de respuesta a las solicitudes de los usuarios sean muy reducidos.

Un sistema gestor de base de datos ayuda a realizar las siguientes acciones:

- ✓ Definición de los datos.
- ✓ Mantenimiento de la integridad de los datos dentro de la base de datos.
- ✓ Control de la seguridad y privacidad de los datos.
- ✓ Manipulación de los datos.

Los SGBD que soporta el CMS Drupal son MySQL y PostgreSQL.

### 1.8.1. PostgreSQL

Es un sistema de gestión de base de datos objeto-relacional para múltiples plataformas. Debido a su licencia liberal, puede usarse, modificarse, y distribuirse por todos gratis para cualquier propósito, sea privado, comercial, o académico. Posee una estabilidad y confiabilidad legendaria, nunca ha presentado caídas en varios años de operación de alta actividad. Tiene un buen soporte para subselects, triggers, vistas y procedimientos almacenados en el servidor, además tiene ciertas características orientadas a objetos.

Como muchos otros proyectos open source, el desarrollo de PostgreSQL no es manejado por una sola compañía sino que es dirigido por una comunidad de desarrolladores y organizaciones comerciales las

cuales trabajan en su desarrollo. Dicha comunidad es denominada el PGDG (PostgreSQL Global Development Group). (3)

### Principales características:

- Incorpora una estructura de datos array.
- Soporta el uso de índices, reglas y vistas.
- Permite la declaración de funciones propias, así como la definición de disparadores.
- Incorpora funciones de diversa índole: manejo de fechas, geométricas, orientadas a operaciones con redes entre otras.
- Incluye herencia entre tablas aunque no entre objetos, ya que no existen, por lo que a este gestor de bases de datos se le incluye entre los gestores objeto-relacionales.
- Permite la gestión de diferentes usuarios, como también los permisos asignados a cada uno de ellos.
- Soporta distintos tipos de datos: además del soporte para los tipos base, también soporta datos de tipo fecha, monetarios, elementos gráficos, datos sobre redes (MAC, IP), cadenas de bits, etc. También permite la creación de tipos propios.

### Ventajas:

- Posee una gran escalabilidad. Es capaz de ajustarse al número de CPUs y a la cantidad de memoria que posee el sistema de forma óptima, haciéndole capaz de soportar una mayor cantidad de peticiones simultáneas de manera correcta (en algunos benchmarks se dice que ha llegado a soportar el triple de carga de lo que soporta MySQL).
- Implementa el uso de rollback's, subconsultas y transacciones, haciendo su funcionamiento mucho más eficaz, y ofreciendo soluciones en campos en las que MySQL no podría.

- Tiene la capacidad de comprobar la integridad referencial, así como también la de almacenar procedimientos en la propia base de datos, equiparándolo con los gestores de bases de datos de alto nivel, como puede ser Oracle.

### 1.9. Servidores web

El servidor web es un programa que corre sobre el servidor que escucha las peticiones HTTP que le llegan y las satisface. Dependiendo del tipo de la petición, el servidor web buscará una página web o bien ejecutará un programa en el servidor. De cualquier modo, siempre devolverá algún tipo de resultado HTML al cliente o navegador que realizó la petición. Va a ser fundamental en el desarrollo de las aplicaciones del lado del servidor que se vaya a construir, ya que se ejecutarán en él.

#### 1.9.1. Apache

Dentro de sus puntos fuertes se encuentran:

- Tiene interfaz de autenticación con todos los sistemas.
- Facilita la integración como "plugins" de lenguajes de programación de páginas web dinámicas.
- Tiene integración en estándar del protocolo de seguridad SSL.
- Provee interfaz a todas las bases de datos.
- Servidor altamente configurable de diseño modular: Se pueden escribir módulos para realizar determinadas funciones, lo que implica que haya gran cantidad de ellos disponibles para su utilización.
- Es flexible, rápido y eficiente.
- Continuamente actualizado y evoluciona a mayor velocidad.
- Multiplataforma.

El Apache fue hecho para proveer un alto grado de calidad y fortaleza para las implementaciones que utilizan el protocolo HTTP. Está ligado a la plataforma (Linux, Windows, UNIX) sobre la cual los individuos o instituciones pueden construir sistemas confiables con fines experimentales o para resolver un problema específico de la organización. (5)

### 1.10. Metodologías de Desarrollo de Software

Para lograr el desarrollo exitoso de proyectos de software cada vez más complejos, surge la necesidad de crear métodos que permitan guiar, controlar y documentar el desarrollo de los mismos. De ahí surgen las metodologías: del griego (metà "más allá" odòs "camino" logos "estudio"). Una metodología es aquella guía que se sigue con el fin de realizar las acciones propias de una investigación. En términos más sencillos se trata de la guía que nos va indicando qué hacer y cómo actuar cuando se quiere obtener algún tipo de investigación. Es posible definir una metodología como aquel enfoque que permite observar un problema de una forma total, sistemática, disciplinada y con cierta disciplina. Las metodologías se pueden clasificar de forma general en:

- **Tradicional (Metodologías Pesadas, o Peso Pesado):** Estas imponen una disciplina de trabajo sobre el proceso de desarrollo del software, con el objetivo de conseguir un software más eficiente y predecible. Para ello, se hace un especial hincapié en la planificación total de todo el trabajo a realizar y una vez que esté todo detallado, comienza el ciclo de desarrollo del producto software.
- **Ágiles (Metodologías Ligeras):** Se caracterizan por tener un desarrollo incremental para producir tempranamente pequeñas entregas en ciclos rápidos, y predisposición para el cambio y la adaptación continua. Estas metodologías por lo general se centran en desarrollar productos funcionales más que en conseguir una buena documentación. (18)

Analizando que la aplicación a desarrollar es una aplicación web, además de la clasificación planteada anteriormente se necesita tener en cuenta a la hora de elegir la metodología a utilizar algunas características que las diferencian de las aplicaciones de escritorio, como es la clasificación de los requerimientos de ese tipo de sistemas, la arquitectura de información, entre otros. Actualmente tres de los más famosos y conocidos procesos de desarrollo de software son:

#### 1.10.1. El Proceso Unificado de Modelado

El Proceso Unificado de Modelado (RUP) es una metodología pesada, está basada en una notación gráfica, la cual permite especificar, construir, visualizar y documentar los artefactos de un sistema de software orientado a objetos. (13) Sus principales características son:

- Guiado por casos de uso: Los casos de uso son el instrumento para describir el comportamiento del software y extraer los casos de prueba con los que se valida el sistema.
- Centrado en la arquitectura: Los modelos son proyecciones del análisis y el diseño, describe la arquitectura del producto a desarrollar.
- Iterativo e incremental: Durante todo el proceso de desarrollo se producen versiones superiores.

RUP utiliza como lenguaje de modelado el Unified Modeling Language (UML) para describir todo el proceso. Se divide en ciclos de trabajo, teniendo un producto superior como resultado de cada ciclo. Estos se componen en su interior por varias fases, en la cuales se llevan a cabo un conjunto de flujos para el desarrollo de todo el proyecto. (15)

### 1.10.2. Programación Extrema

La metodología Programación Extrema o Extreme Programming (XP), es conocida como metodología ágil o ligera, orientada al cliente, y de iteraciones cortas. La base para el desarrollo del software que usa esta metodología son las llamadas historias de usuario escritas por el cliente en las que describen escenarios sobre el funcionamiento del sistema y que no sólo están limitados los formularios de usuario, sino que también pueden describir modelos. (10)

Estas historias de usuario junto a la arquitectura que se persigue, sirven de base para crear un plan de “entregas de software” entre el equipo de desarrollo y el cliente. Propone que en el equipo de desarrollo se necesita un representante constante del cliente que conozca al dedillo el negocio y que esté a disposición para cualquier duda o necesidades que los desarrolladores tengan. El cliente se mantiene todo el tiempo informado paso por paso de las cosas que se están desarrollando, y a medida que se dé la liberación de cualquier entregable se discutirá con el representante, y se repite la nueva iteración del software.

La programación del software siempre se define en pareja con el objetivo principal de lograr mayores resultados y los menores errores posibles, mientras uno codifica haciendo hincapié en la calidad de la función o método que está implementando, el otro analiza si ese método o función es adecuado y está bien diseñado. Es necesario que los desarrolladores se reúnan diariamente y expongan sus problemas, soluciones e ideas de forma conjunta. (11)

### 1.10.3. Desarrollo Guiado por la Funcionalidad

El Desarrollo Guiado por la Funcionalidad (FDD) es un proceso que se puede considerar a medio camino entre RUP y XP, aunque es más similar al último. Está pensado para proyectos de corto tiempo de duración (menos de un año), un proceso iterativo con iteraciones cortas (aproximadamente 2 semanas). Las iteraciones se deciden en base a las funcionalidades, que son pequeñas partes del software con significado para el cliente.

**Un proyecto con FDD se divide en 5 fases:**

1. Desarrollo de un modelo general.
2. Construcción de la lista de funcionalidades.
3. Plan de releases en base a las funcionalidades a implementar.
4. Diseñar en base a las funcionalidades.
5. Implementar en base a las funcionalidades.

El trabajo se realiza en grupo aunque siempre habrá un responsable (arquitecto jefe o jefe de programadores, en dependencia de la fase en la que se encuentre) con mayor experiencia. Las funcionalidades se dividen entre los subgrupos del equipo. FDD también define métricas para seguir el proceso de desarrollo de la aplicación.

**Puntos claves:** Ligero, a medio camino entre el desarrollo y la organización, existe una jerarquía dentro del equipo, el código fuente tiene propietario, los equipos varían en función de la funcionalidad a implementar, el conocimiento de la aplicación se reparte a través del trabajo en equipo y revisiones. Documentación aceptable.

### 1.11. Herramientas utilizadas para el desarrollo del sistema

#### 1.11.1. Visual Paradigm

Visual Paradigm es una herramienta CASE (Ingeniería de Software Asistida por Ordenador, en inglés Computer Aided Software Engineering) que utiliza UML como lenguaje de modelado. Soporta el ciclo de vida completo de desarrollo de un software, desde la fase de análisis hasta el despliegue del mismo. Permite realizar ingeniería directa o inversa sobre el software, es capaz a partir de un modelo relacional

en diferentes SGBD, desplegar todas las clases asociadas a las tablas y soporta múltiples usuarios trabajado sobre el mismo proyecto.

### 1.11.2. Zend Studio

Zend Studio es uno de los ambientes de desarrollo integrado o Integrated Development Environment (IDE) disponible para desarrolladores profesionales que agrupa todos los componentes de desarrollo necesarios para ciclo de desarrollo de aplicaciones PHP. A través de un comprensivo conjunto de herramientas de edición, depurado, análisis, optimización y bases de datos, Zend Studio acelera los ciclos de desarrollo y simplifica los proyectos complejos.

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

### 1.12. Fundamentación de las herramientas seleccionadas

Después de realizar un estudio de las tendencias actuales y de las principales características de las tecnologías más usadas en la actualidad, se propone utilizar un CMS para la creación de la aplicación, específicamente el Drupal, debido a su rapidez de respuesta ante la petición de algún usuario, posee independencia de la base de datos y es el CMS más seguro. PHP, como es el lenguaje de programación en el que está creado Drupal, necesita ser instalado para que pueda ser interpretado por el servidor, además de que es multiplataforma, posee una sintaxis familiar para el desarrollo de la aplicación y cuenta con una amplia comunidad de programadores, no solo en Internet, sino en la propia UCI, lo que permite acceder fácilmente a su documentación por parte de sus usuarios.

Como sistema gestor de base de datos PostgreSQL, pues es un sistema objeto-relacional, que incluye características de la orientación a objetos, como son la herencia, tipos de datos, funciones, restricciones, disparadores, reglas e integridad transaccional. Se escogió el servidor Apache, ya que es un servidor multiplataforma, rápido y eficiente y puede ser adaptado a diferentes entornos y necesidades, con los diferentes módulos de apoyo que proporciona.

Como metodología de desarrollo XP debido a que se adapta en gran medida, tanto al tipo de proyecto a desarrollar como a las condiciones de trabajo y el cliente forma parte del equipo de desarrollo. Mediante la aplicación de XP se puede lograr una retroalimentación mayor y lograr un producto que satisfaga sus

necesidades. Como herramienta de modelado en UML el Visual Paradigm. Para apoyar el proceso de implementación del software se escogió la herramienta de desarrollo Zend Studio.

Otra de las razones para la selección de las herramientas a utilizar, es que además de todas las ventajas mencionadas anteriormente, estas son las herramientas que se ajustan a la arquitectura para los sistemas que plantea la Dirección de Informatización de UCI, perteneciente a la Infraestructura Productiva (IP) en la UCI.

### **1.13. Conclusiones**

Como resultado del análisis realizado durante el presente capítulo, se pudo concluir que: ninguno de los portales estudiados, relacionados con la gestión de información sobre calidad del software, responden a las características de este proceso en la facultad 1, por lo que se necesita diseñar un nuevo portal que dinamice el proceso mencionado con las herramientas seleccionadas.



# CAPÍTULO 2: CARACTERÍSTICAS DEL SISTEMA

### 2.1. Introducción

El presente capítulo tiene como objetivo hacer una valoración de las características principales del portal a desarrollar, se especifica detalladamente los requerimientos funcionales y no funcionales, además se hace alusión a las fases de exploración y planificación, propias de la metodología de desarrollo utilizada para la implementación del sistema que se propone, donde se exponen los artefactos generados durante el transcurso de las mismas.

### 2.2. Conceptos sobre Calidad de Software

“Grado con el que un sistema, componente o proceso cumple los requerimientos especificados y las necesidades o expectativas del cliente o usuario”. (20)

“Concordancia del software producido con los requerimientos explícitamente establecidos, con los estándares de desarrollo prefijados y con los requerimientos implícitos no establecidos formalmente, que desea el usuario”.(15)

### 2.3. Arquitectura de la Información

La Arquitectura de Información es el arte de estructurar los sitios web o portales a través de un conjunto de métodos y herramientas con el fin de ayudar a los usuarios a encontrar y manejar la información. Comprende los sistemas de organización y estructuración de los contenidos y los sistemas de rotulado o etiquetado de los mismos. De esta manera el usuario puede encontrar, utilizar y comprender la información.

Para realizar la Arquitectura de Información del Portal de Calidad de la Facultad 1 se tuvo en cuenta elementos como los objetivos del portal, la definición de la audiencia y los contenidos y servicios, su estructura y el sistema de navegación, logrando así que el contenido del portal quede organizado.

#### 2.3.1. Objetivos

El Portal de Calidad de la Facultad 1 es de vital importancia para establecer una guía o patrón para todos los proyectos de la facultad, este tiene como objetivos fundamentales:

- Establecer una interrelación más fluida entre el asesor, los aseguradores de calidad de cada uno de los proyectos y los jefes de los mismos, mediante un foro de debate sobre un tema determinado.
- Crear las condiciones para que los usuarios tengan un medio de fácil obtención de información.
- Mostrar los elementos por los que se deben guiar de forma obligatoria todos los proyectos de la facultad.
- Publicar cursos de formación para los aseguradores de calidad de cada proyecto.
- Gestionar todo el proceso de revisiones y las auditorías de cada uno de los proyectos.

### 2.3.2. Definición de Audiencia

La audiencia no es más que el público hacia el cual estará orientado el producto. En el Portal de Calidad de Software esta se dividió en dos grupos fundamentales.

#### 1. Personal que mantiene el portal.

- Administradores: Usuarios con pleno acceso a todos los elementos del portal

#### 2. Personas que utilizan el portal como medio de información.

- Usuario autenticado: Personas con conocimiento del portal, que entran a buscar un contenido determinado.
- Usuario anónimo: Persona que entra al portal en la búsqueda de alguna información, pero sin conocimiento de si existe o no.
- Vicedecano de producción: Es la persona facultada para ver información referente a los proyectos.
- Asesor de Calidad: Es la persona facultada para realizar el cronograma de revisiones internas a los proyectos y llevar a cabo todo el proceso de las mismas al igual que realizar las auditorías.
- Jefe de Polo: Es la persona facultada para ver información referente a los proyectos.
- Líder de proyecto: Es la persona facultada para gestionar información referente al expediente de su proyecto.
- Asegurador de calidad: Es la persona facultada para evaluar la información referente al expediente de proyecto.

El portal está dirigido hacia profesores y estudiantes de la UCI con necesidades de obtención de información en el tema de calidad de software y sobre todo para la facultad 1.

### 2.3.3. Definición de contenidos y servicios del portal

Luego de haber identificado los objetivos y la audiencia, se pueden definir los contenidos y los servicios del sitio. Estos se pueden agrupar en:

- Inicio.
- Preguntas Frecuentes.
- Miembros.
- Cronograma de Revisiones.
- Enviar y recibir solicitud de revisión
- Insertar resultado de auditoría y revisiones
- Reportes
- Mapa de navegación.
- Información de proyectos.
- Publicar cursos de formación.
- Documentación.
- Eventos.
- Noticias.
- Encuestas.
- Enlaces.
- Fórum.
- Calendario.

### 2.3.4. Definición de la estructura del portal

Antes de entrar en detalle, es necesario definir los conceptos de estructura y diseño, a modo de aclaración, debido a que estos términos tienden a confundir:

- **Estructura:** se refiere a la forma que tendrá el portal web en términos generales con sus secciones, funcionalidades y sistemas de navegación. No considera ni incluye elementos gráficos (logotipos, viñetas, etc.).
- **Diseño:** se refiere a la solución gráfica que se creará para el portal, en la cual aparecen colores, logotipos, viñetas, y otros elementos de diseño que permiten identificar visualmente al portal.

### 2.3.4.1. Estructura del Portal de Calidad de la Facultad 1

1. Inicio.

1.1. Navegación.

1.1.1. Preguntas Frecuentes.

1.1.2. Cronograma de revisiones.

1.1.3. Miembros.

1.1.4. Enviar solicitud de revisión.

1.1.5. Recibir solicitud de revisión.

1.1.6. Insertar resultado de las revisiones.

1.1.7. Insertar resultado de las auditorías.

1.1.8. Reportes.

1.1.9. Publicar cursos de formación.

1.2. Documentación.

1.3. Noticias.

1.4. Fórum.

1.5. Encuestas.

1.6. Enlaces.

1.7. Buscar.

1.8. Calendario.

1.9. Información de proyectos.

1.10. Mapa de navegación.

### 2.3.5. Mapa de navegación del Portal de Calidad

Es una representación visual de forma jerárquica que muestra la estructura general del portal. A continuación se muestra el mapa conceptual del Portal de Calidad.

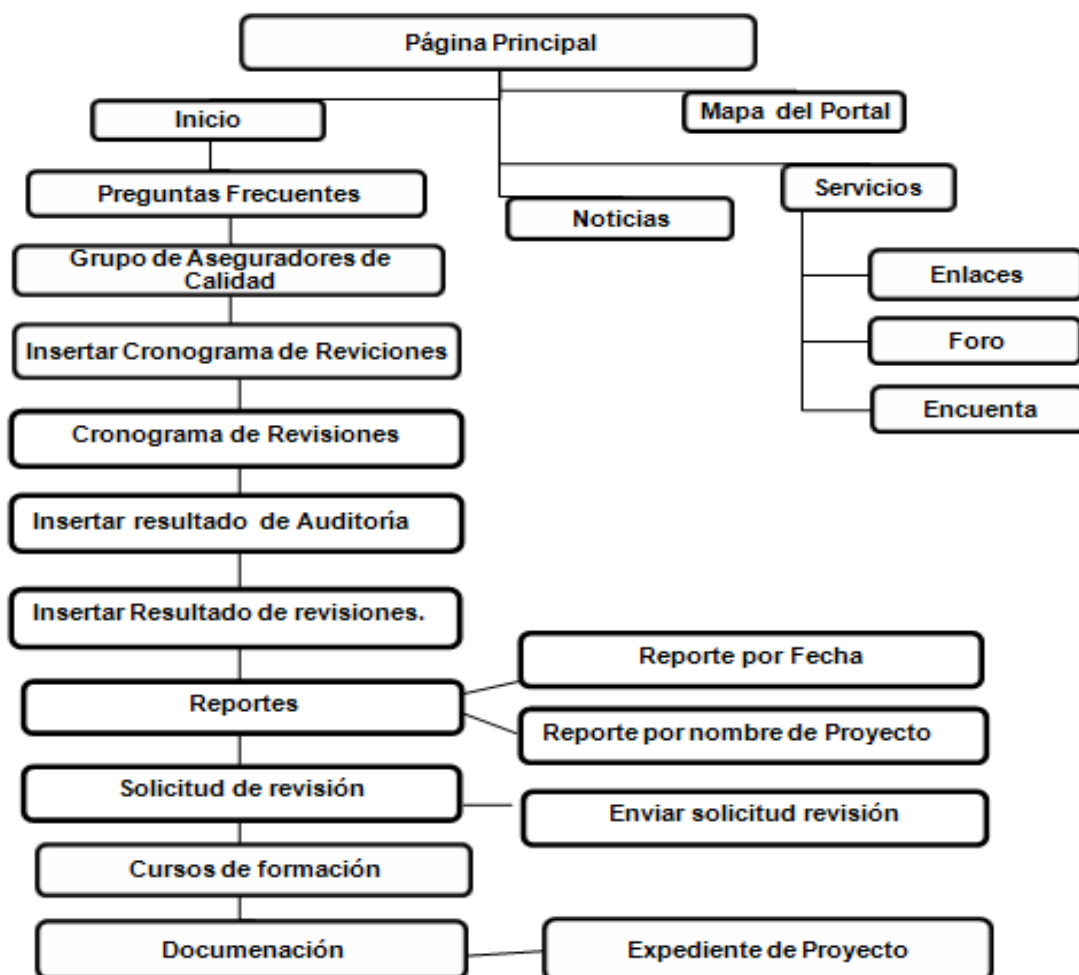


Figura 1 Mapa de Navegación del Portal de Calidad.

### 2.3.6. Elementos de los Sistemas de Navegación

- **Menú General:** Siempre presente en todo el sitio, permite el acceso a cada una de las áreas del sitio.
- **Pié de Página:** Usualmente ubicado en la parte inferior de cada página, indica el nombre de la institución, teléfonos, dirección física y de correo electrónico.
- **Fecha de publicación:** Para saber la vigencia de publicación del contenido desplegado.
- **Botón Inicio:** Para ir a la portada.
- **Botón Mapa del sitio:** Para ver el mapa del sitio web.
- **Buscador:** Presente en cada página si es que la funcionalidad existe en el sitio.

- **Botón Imprimir:** Para imprimir el contenido de la página; se espera que el formato de impresión del documento que se muestra en pantalla sea más simple que la página normal del sitio web, para dar la impresión al usuario de que hay una preocupación por ayudarle en la tarea de llevar impreso el contenido.

### 2.3.7. Diseño de la estructura de la página principal

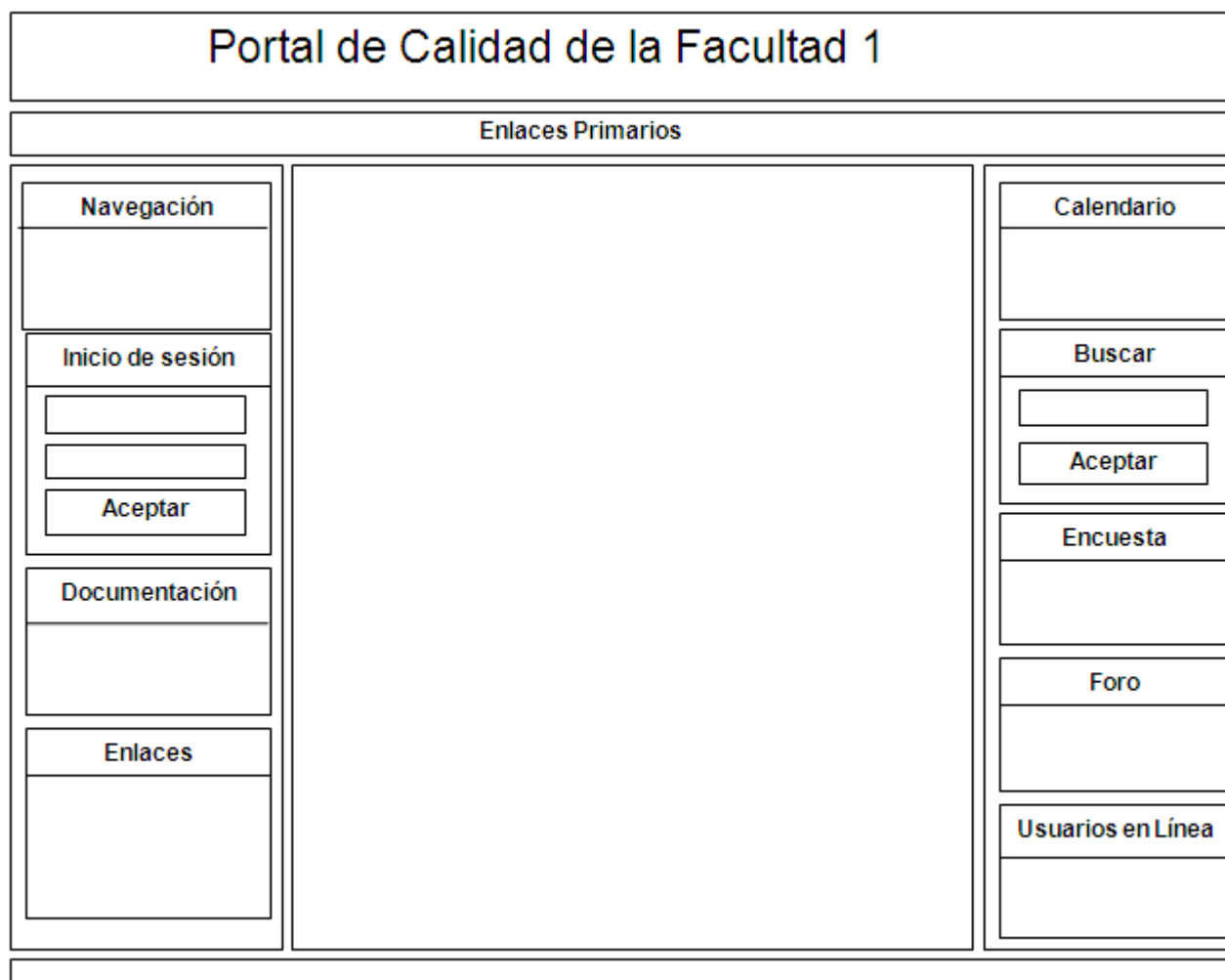


Figura 2 Diseño de la estructura de la página principal.

### 2.4. Requerimientos funcionales del sistema

R1 Autenticar.

R2 Gestionar documentación.

R2.1 Adicionar documento.

R2.2 Mostrar documento.

R2.3 Modificar documento.

R2.4 Eliminar documento.

R3 Gestionar información de proyectos.

R3.1 Adicionar información de proyectos.

R3.2 Mostrar información de proyectos.

R3.3 Modificar información de proyectos.

R3.4 Eliminar información de proyectos.

R4 Gestionar miembros por proyectos.

R4.1 Mostrar miembros por proyectos.

R4.2 Modificar miembros por proyectos.

R4.3 Eliminar miembros por proyectos.

R5 Gestionar noticias.

R 5.1 Adicionar noticia.

R 5.2 Modificar noticia.

R 5.3 Eliminar noticia.

R 5.4 Mostrar Noticias.

R6 Gestionar fórum.

R6.1 Añadir fórum.

R6.2 Editar fórum.

R6.3 Eliminar fórum.

R7 Gestionar encuesta.

R7.1 Adicionar encuesta.

R7.2 Eliminar encuesta.

R7.3 Mostrar encuesta.

R7.4 Modificar encuesta.

R8 Gestionar enlaces.

R8.1 Adicionar enlaces.

R8.2 Eliminar enlaces.

R8.3 Mostrar enlaces con otros sitios.

R9 Buscar información.

R10 Imprimir documentos.

R11 Autenticar contra dominio LDAP.

R12 Gestionar cronograma de revisiones.

R12.1 Adicionar cronograma de revisiones.

R12.2 Eliminar cronograma de revisiones.

R12.3 Mostrar cronograma de revisiones.

R12.4 Modificar cronograma de revisiones.

R13 Gestionar preguntas frecuentes.

R13.1 Adicionar preguntas frecuentes.

R13.2 Eliminar preguntas frecuentes.

R13.3 Mostrar preguntas frecuentes.

R13.4 Modificar preguntas frecuentes.

R14 Mostrar el mapa de navegación.

R15 Gestionar solicitud de revisión.

R17.1 Enviar solicitud de revisión.



R17.2 Mostrar solicitud de revisión.

R16 Insertar resultado de revisión.

R17 Mostrar reporte de resultado de revisión.

R18 Insertar resultado de auditoría.

R19Mostrar reporte resultado de auditoría.

R20 Gestionar eventos.

R20.1 Adicionar eventos.

R20.2 Eliminar eventos.

R20.3 Mostrar eventos.

R20.4 Modificar eventos.

R21 Mostrar cantidad de no conformidades a jefes de proyectos.

R22 Publicar cursos de formación.

### **2.5. Requerimientos no funcionales del sistema**

Los requerimientos no funcionales son propiedades o cualidades que el producto debe tener. Debe pensarse en estas propiedades como las características que hacen al producto atractivo, usable, rápido o confiable.

#### **Apariencia o interfaz externa:**

- Diseño perfectamente encuadrado para resoluciones de 800x600, pero preparado para verse en otras resoluciones.
- Diseño gráfico acorde con las pautas de diseño de la Universidad.

#### **Usabilidad:**

El sistema podrá ser usado por cualquier persona que posea conocimientos básicos en el manejo de la computadora y de un ambiente web en sentido general.

### Seguridad:

- Identificar al usuario antes de que pueda realizar cualquier acción sobre el sistema.
- Proteger la información manejada por el sistema de accesos no autorizados.
- Garantizar que las funcionalidades del sistema se muestren de acuerdo al nivel de usuario que esté activo. Esto se logra según el tipo de usuario y su rol.
- Proteger la información manejada por el sistema contra la corrupción de ficheros o estados inconsistentes.
- Verificación sobre acciones irreversibles (eliminaciones).

### Legales:

El CMS escogido como plataforma para el desarrollo de la aplicación está basado en la licencia GNU/GPL.

### Soporte:

- Sistema multiplataforma.
- Gestor de base de datos con soporte para grandes volúmenes de datos y alta velocidad de procesamiento. Tiempo de respuesta rápido en accesos concurrentes.

### Software:

- Aplicación web escrita sobre el lenguaje de programación PHP 5.2.5.
- Usar el gestor de Base de Datos PostgreSQL 8.2.4.
- Utilizar como servidor web Apache 2.2.6.
- Desarrollada bajo CMS Drupal 5.15.

### Hardware:

Se requiere de una computadora con 128 MB de memoria RAM como mínimo, con un servidor web de 512 MB de RAM y con una capacidad mínima de disco duro de 20 GB. Mouse y teclado.

### 2.6. Personas relacionadas con el sistema

Se define como persona relacionada al sistema toda aquella que obtiene un resultado de valor de uno o varios procesos que se ejecutan en el mismo. Además de aquellas que se encuentran involucradas en dichos procesos, pues participan en ellos pero no obtienen ningún resultado de valor.

**Tabla 2. 1 Personas relacionadas con el sistema**

<b>Personas relacionadas con el sistema</b>	<b>Justificación</b>
Usuario anónimo	Es la persona que navega por el sistema sin haberse creado una cuenta aún, fluye dentro de éste sin privilegios. Tiene la posibilidad de crearse una cuenta, ver su perfil, navegar sobre las diferentes opciones para irse relacionando con el mismo, así como indagar y descargar, en caso que desee, los diferentes documentos que se muestran de manera general.
Usuario autenticado	Es la persona que se encuentra autenticada en el sitio cumpliendo este rol, además tendrá acceso a diferentes módulos debido al rol que cumple dentro de la aplicación.
Administrador	Es la persona facultada para la gestión del sistema y el encargado de administrar las diferentes cuentas de los usuarios autenticados en la aplicación.
Vicedecano de producción	Es la persona facultada para ver información referente a los proyectos.
Asesor de Calidad	Es la persona facultada para realizar el cronograma de revisiones internas a los proyectos y llevar a cabo todo el proceso de las mismas al igual de realizar las auditorías.
Jefe de Polo	Es la persona facultada para ver información referente a los proyectos.
Líder de proyecto	Es la persona facultada para gestionar información referente al expediente de su proyecto y ver la misma.
Asegurador de calidad	Es la persona facultada para evaluar la información referente al expediente de proyecto.

### 2.7. Aplicación de la metodología seleccionada

#### 2.7.1. Fase de exploración

La metodología de desarrollo Programación Extrema (XP) comienza con su fase de exploración. Durante esta etapa se realiza el proceso de identificación de las historias de usuario (UH, del inglés User Histories), así como la familiarización de los equipos de trabajo con las tecnologías y herramientas seleccionadas para la construcción del sistema.

El ciclo de vida de XP enfatiza en el carácter iterativo e incremental del desarrollo. Una iteración de desarrollo es un período de tiempo en el que se realiza un conjunto de funcionalidades determinadas, que en el caso de XP corresponden a un conjunto de historias de usuarios. (12)

##### 2.7.1.1. Historias de usuarios

Uno de los artefactos más importantes que genera la metodología XP son las historias de usuario (HU), pues son la forma en que se especifican en XP los requisitos del sistema. Estas se redactan desde la perspectiva del cliente aunque los desarrolladores pueden brindar también su ayuda en la identificación de las mismas. El contenido que ellas abarcan debe ser concreto y sencillo.

Las HU conducen al proceso de creación de los test de aceptación, los cuales servirán para verificar que estas historias se han implementado correctamente. Otra de sus características es que solamente proporcionan los detalles sobre la estimación del riesgo y cuánto tiempo conllevará su implementación. Su nivel de detalle debe ser el mínimo posible, de manera que permita hacerse una ligera idea de cuánto costará implementar el sistema. (12)

Durante este proceso se identificaron 22 historias de usuarios, cada una de ellas respondiendo a las diferentes funcionalidades solicitadas por el cliente y dando una idea al resto del equipo de desarrollo de cómo debe ser su posterior implementación, las cuales se detallan a continuación:

**Tabla 2. 2 HU Autenticar**

**Historia de Usuario**

<b>Número: 1</b>	<b>Nombre: Autenticar</b>
<b>Usuario:</b> Crear los siguientes usuarios y roles Usuario Autenticado y Administrador, Vicedecano de producción, Jefe de Polo, Líder de proyecto, Asesor de calidad y Asegurador de calidad.	
<b>Prioridad en negocio: Alta</b>	<b>Riesgo en Desarrollo: Alto</b>
<b>Puntos de Estimación: 1</b>	<b>Iteración Asignada: 1</b>
<b>Descripción:</b> Se brinda la posibilidad de que la persona que acceda al sistema introduzca sus datos (usuario y contraseña) con la finalidad de verificar y otorgarle los permisos según el rol que cumpla dentro de la aplicación.	

**Observaciones:** Se hace referencia al requisito R1.

**Tabla 2. 3 HU Gestionar documentación**

Historia de Usuario	
<b>Número: 2</b>	<b>Nombre:</b> Gestionar documentación
<b>Usuario:</b> Administrador	
<b>Prioridad en negocio:</b> Alta	<b>Riesgo en Desarrollo:</b> Alto
<b>Puntos de Estimación: 2</b>	<b>Iteración Asignada: 1</b>
<b>Descripción:</b> Se brinda la posibilidad al administrador de adicionar, modificar, eliminar y mostrar en el sistema los documentos.	
<b>Observaciones:</b> Se hace referencia al requisito R2.	

**Tabla 2. 4 HU Gestionar información de proyectos**

Historia de Usuario	
<b>Número: 3</b>	<b>Nombre:</b> Gestionar información de proyectos
<b>Usuario:</b> Líder de proyecto	
<b>Prioridad en negocio:</b> Alta	<b>Riesgo en Desarrollo:</b> Alto
<b>Puntos de Estimación: 1</b>	<b>Iteración Asignada: 1</b>
<b>Descripción:</b> Se brinda la posibilidad al Líder de proyecto de adicionar, modificar, eliminar y mostrar en el sistema la información de proyectos.	
<b>Observaciones:</b> Se hace referencia al requisito R3.	

**Tabla 2. 5 HU Gestionar listado de miembros por proyectos**

Historia de Usuario	
<b>Número: 4</b>	<b>Nombre:</b> Gestionar miembros por proyectos
<b>Usuario:</b> Administrador y Asesor de Calidad.	
<b>Prioridad en negocio:</b> Media	<b>Riesgo en Desarrollo:</b> Medio
<b>Puntos de Estimación: 1</b>	<b>Iteración Asignada: 2</b>
<b>Descripción:</b> Se brinda la posibilidad al administrador de modificar, eliminar y mostrar en el sistema los miembros por proyectos.	
<b>Observaciones:</b> Se hace referencia al requisito R4.	

**Tabla 2. 6 HU Gestionar noticias**

Historia de Usuario	
<b>Número: 5</b>	<b>Nombre:</b> Gestionar noticia
<b>Usuario:</b> Administrador	
<b>Prioridad en negocio:</b> Media	<b>Riesgo en Desarrollo:</b> Medio
<b>Puntos de Estimación: 1</b>	<b>Iteración Asignada: 2</b>
<b>Descripción:</b> Se brinda la posibilidad al administrador de adicionar, modificar, eliminar y mostrar en el sistema una noticia.	
<b>Observaciones:</b> Se hace referencia al requisito R5.	

**Tabla 2. 7 HU Gestionar Fórum**

Historia de Usuario	
<b>Número:6</b>	<b>Nombre:</b> Gestionar Fórum
<b>Usuario:</b> Administrador.	
<b>Prioridad en Negocio:</b> Medio	<b>Riesgo en Desarrollo:</b> Medio
<b>Puntos de Estimación:</b> 1	<b>Iteración Asignada:</b> 2
<b>Descripción:</b> Se brinda la posibilidad al administrador de añadir, editar, mostrar y eliminar un fórum.	
<b>Observaciones:</b> Se hace referencia al requisito R6	

**Tabla 2. 8 HU Gestionar Encuesta**

Historia de Usuario	
<b>Número:7</b>	<b>Nombre:</b> Gestionar Encuesta
<b>Usuario:</b> Administrador.	
<b>Prioridad en Negocio:</b> Medio	<b>Riesgo en Desarrollo:</b> Medio
<b>Puntos de Estimación:</b> 1	<b>Iteración Asignada:</b> 2
<b>Descripción:</b> Se brinda la posibilidad al administrador de adicionar, modificar, mostrar, eliminar una encuesta, relacionada para medir la utilidad del sistema.	
<b>Observaciones:</b> Se hace referencia al requisito R7.	

**Tabla 2. 9 HU Gestionar Enlaces**

Historia de Usuario	
<b>Número:8</b>	<b>Nombre:</b> Gestionar Enlace
<b>Usuario:</b> Administrador.	
<b>Prioridad en Negocio:</b> Medio	<b>Riesgo en Desarrollo:</b> Medio
<b>Puntos de Estimación:</b> 1	<b>Iteración Asignada:</b> 2
<b>Descripción:</b> Se brinda la posibilidad al administrador de adicionar, mostrar, eliminar un enlace en el sistema.	
<b>Observaciones:</b> Se hace referencia al requisito R8.	

**Tabla 2. 10 HU Buscar Información**

Historia de Usuario	
<b>Número: 9</b>	<b>Nombre:</b> Buscar información
<b>Usuario:</b> Usuario Anónimo y Autenticado, Administrador, Vicedecano de producción, Jefe de Polo, Líder de proyecto, Asesor de calidad y Asegurador de calidad.	
<b>Prioridad en negocio:</b> Baja	<b>Riesgo en Desarrollo:</b> Bajo
<b>Puntos de Estimación:</b> 1	<b>Iteración Asignada:</b> 3
<b>Descripción:</b> Se brinda la posibilidad de que cualquier persona que acceda a la aplicación tenga la posibilidad de buscar contenidos que se encuentran dentro del sistema.	

**Observaciones:** Se hace referencia al requisito R9.

**Tabla 2. 11 HU Imprimir documentos**

Historia de Usuario	
<b>Número: 10</b>	<b>Nombre:</b> Imprimir documentos
<b>Usuario:</b> Usuario Anónimo y Autenticado, Administrador, Vicedecano de producción, Jefe de Polo, Líder de proyecto, Asesor de calidad y Asegurador de calidad.	
<b>Prioridad en negocio:</b> Baja	<b>Riesgo en Desarrollo:</b> Bajo
<b>Puntos de Estimación: 1</b>	<b>Iteración Asignada: 3</b>
<b>Descripción:</b> Se brinda la posibilidad dependiendo del rol, de la impresión de los documentos en el sistema.	
<b>Observaciones:</b> Se hace referencia al requisito R10.	

**Tabla 2. 12 HU Autenticar contra dominio LDAP**

Historia de Usuario	
<b>Número: 11</b>	<b>Nombre:</b> Autenticar contra dominio LDAP
<b>Usuario:</b> Administrador	
<b>Prioridad en Negocio:</b> Bajo	<b>Riesgo en Desarrollo:</b> Alto
<b>Puntos de Estimación: 1</b>	<b>Iteración Asignada: 1</b>
<b>Descripción:</b> El sistema utiliza un dominio LDAP existente. De esta forma es más fácil usarlo en intranet.	
<b>Observaciones:</b> Esta opción se desarrolló debido a las circunstancias de contar con un dominio LDAP. Su principal ventaja en una intranet cerrada es la posibilidad de evitar duplicación de usuarios y proliferación de los mismos.	

**Tabla 2. 13 HU Gestionar cronograma de revisiones**

Historia de Usuario	
<b>Número: 12</b>	<b>Nombre:</b> Gestionar cronograma de revisiones
<b>Usuario:</b> Administrador y Asesor de calidad.	
<b>Prioridad en negocio:</b> Baja	<b>Riesgo en Desarrollo:</b> Alto
<b>Puntos de Estimación: 1</b>	<b>Iteración Asignada: 1</b>
<b>Descripción:</b> Se brinda la posibilidad al administrador y al asesor de calidad de adicionar, modificar, eliminar y mostrar el cronograma de revisiones para los proyectos en el sistema.	
<b>Observaciones:</b> Se hace referencia al requisito R12.	

**Tabla 2. 14 HU Gestionar preguntas frecuentes**

Historia de Usuario	
<b>Número: 13</b>	<b>Nombre:</b> Gestionar preguntas frecuentes
<b>Usuario:</b> Administrador.	
<b>Prioridad en negocio:</b> Baja	<b>Riesgo en Desarrollo:</b> Bajo
<b>Puntos de Estimación: 1</b>	<b>Iteración Asignada: 3</b>

<b>Descripción:</b> Se brinda la posibilidad al administrador de adicionar, modificar, eliminar y mostrar las preguntas frecuentes en el sistema.
<b>Observaciones:</b> Se hace referencia al requisito R13.

**Tabla 2. 15 HU Mostrar el mapa de navegación**

Historia de Usuario	
<b>Número: 14</b>	<b>Nombre:</b> Mostrar el mapa de navegación
<b>Usuario:</b> Administrador	
<b>Prioridad en negocio:</b> Baja	<b>Riesgo en Desarrollo:</b> Bajo
<b>Puntos de Estimación: 1</b>	<b>Iteración Asignada: 1</b>
<b>Descripción:</b> Se brinda la posibilidad al administrador de mostrar el mapa de navegación del portal de calidad en el sistema.	
<b>Observaciones:</b> Se hace referencia al requisito R14.	

**Tabla 2. 16 HU Gestionar solicitud de revisión**

Historia de Usuario	
<b>Número: 15</b>	<b>Nombre:</b> Gestionar solicitud de revisión
<b>Usuario:</b> Líder de proyecto y Asesor de calidad	
<b>Prioridad en negocio:</b> Baja	<b>Riesgo en Desarrollo:</b> Alto
<b>Puntos de Estimación: 1</b>	<b>Iteración Asignada: 1</b>
<b>Descripción:</b> Se brinda la posibilidad al Líder de proyecto enviar una solicitud de revisión y al Asesor de calidad poder recibir las mismas en el sistema.	
<b>Observaciones:</b> Se hace referencia al requisito R15.	

**Tabla 2. 17 HU Insertar resultado de revisión**

Historia de Usuario	
<b>Número: 16</b>	<b>Nombre:</b> Insertar resultado de revisión
<b>Usuario:</b> Asesor de calidad	
<b>Prioridad en negocio:</b> Baja	<b>Riesgo en Desarrollo:</b> Alto
<b>Puntos de Estimación: 1</b>	<b>Iteración Asignada: 1</b>
<b>Descripción:</b> Se brinda la posibilidad al Asesor de calidad poder insertar el resultado de las revisiones en el sistema.	
<b>Observaciones:</b> Se hace referencia al requisito R16.	

**Tabla 2. 18 HU Mostrar reporte de resultado de revisión.**

Historia de Usuario	
<b>Número: 17</b>	<b>Nombre:</b> Mostrar reporte de resultado de revisión
<b>Usuario:</b> Asesor de calidad y Líder de proyecto	
<b>Prioridad en negocio:</b> Baja	<b>Riesgo en Desarrollo:</b> Alto
<b>Puntos de Estimación: 1</b>	<b>Iteración Asignada: 1</b>
<b>Descripción:</b> Se brinda la posibilidad al Asesor de calidad y al Líder de proyecto poder ver el resultado de las revisiones mediante un reporte en el sistema.	



**Observaciones:** Se hace referencia al requisito R17.

**Tabla 2. 19 HU Insertar resultado de auditoría.**

Historia de Usuario	
<b>Número: 18</b>	<b>Nombre:</b> Insertar resultado de auditoría
<b>Usuario:</b> Asesor de calidad	
<b>Prioridad en negocio:</b> Baja	<b>Riesgo en Desarrollo:</b> Alto
<b>Puntos de Estimación: 1</b>	<b>Iteración Asignada: 1</b>
<b>Descripción:</b> Se brinda la posibilidad al Asesor de calidad poder insertar resultado de auditoría en el sistema.	
<b>Observaciones:</b> Se hace referencia al requisito R218.	

**Tabla 2. 20 HU Mostrar reporte de resultado de auditoría.**

Historia de Usuario	
<b>Número: 19</b>	<b>Nombre:</b> Mostrar reporte de resultado de auditoría
<b>Usuario:</b> Asesor de calidad y Líder de proyecto	
<b>Prioridad en negocio:</b> Baja	<b>Riesgo en Desarrollo:</b> Alto
<b>Puntos de Estimación: 1</b>	<b>Iteración Asignada: 1</b>
<b>Descripción:</b> Se brinda la posibilidad al Asesor de calidad y al Líder de proyecto poder ver el resultado de las auditorías mediante un reporte en el sistema.	
<b>Observaciones:</b> Se hace referencia al requisito R19.	

**Tabla 2. 21 HU Gestionar eventos**

Historia de Usuario	
<b>Número: 20</b>	<b>Nombre:</b> Gestionar eventos
<b>Usuario:</b> Administrador	
<b>Prioridad en negocio:</b> Baja	<b>Riesgo en Desarrollo:</b> Bajo
<b>Puntos de Estimación: 1</b>	<b>Iteración Asignada: 3</b>
<b>Descripción:</b> Se brinda la posibilidad al Administrador poder adicionar, modificar, eliminar y mostrar los eventos en el sistema.	
<b>Observaciones:</b> Se hace referencia al requisito R20.	

**Tabla 2. 22 HU Mostrar cantidad de no conformidades a los jefes de proyectos**

Historia de Usuario	
<b>Número: 21</b>	<b>Nombre:</b> Mostrar cantidad de no conformidades a los jefes de proyectos.
<b>Usuario:</b> Administrador	
<b>Prioridad en negocio:</b> Baja	<b>Riesgo en Desarrollo:</b> Bajo
<b>Puntos de Estimación: 1</b>	<b>Iteración Asignada: 3</b>
<b>Descripción:</b> Se brinda la posibilidad al Administrador poder mostrar la cantidad de no conformidades a los jefes de proyectos en el sistema.	
<b>Observaciones:</b> Se hace referencia al requisito R21.	

**Tabla 2. 23 HU Publicar cursos de formación.**

Historia de Usuario	
<b>Número: 22</b>	Nombre: Publicar cursos de formación.
<b>Usuario:</b> Administrador y Asesor de calidad	
<b>Prioridad en negocio:</b> Baja	<b>Riesgo en Desarrollo:</b> Medio
<b>Puntos de Estimación:</b> 1	<b>Iteración Asignada:</b> 2
<b>Descripción:</b> Se brinda la posibilidad al Administrador y al Asesor de calidad poder publicar cursos de formación para los aseguradores de calidad en el sistema.	
<b>Observaciones:</b> Se hace referencia al requisito R22.	

### 2.7.2. Fase de planificación

La metodología XP plantea la planificación como un diálogo continuo entre las partes involucradas en el proyecto, incluyendo al cliente, a los programadores y a los coordinadores o gerentes. Es una fase corta, en la que el cliente, los gerentes y el grupo de desarrolladores acuerdan el orden en que deberán implementarse las historias de usuario, y, asociadas a éstas, las entregas. Típicamente esta fase consiste en una o varias reuniones grupales de planificación. El resultado de esta fase es un Plan de Entregas.

La planificación no debe ser estricta puesto que hay muchas variables en juego, debe ser flexible para poder adaptarse a los cambios que puedan surgir. Una buena estrategia es hacer planificaciones detalladas para unas pocas semanas y planificaciones mucho más abiertas para unos pocos meses.

#### 2.7.2.1. Estimación de esfuerzos por historias de usuario

Durante la fase de planificación se realiza una estimación del esfuerzo que costará implementar cada historia de usuario. Esto se expresa utilizando como medida el punto. Un punto se considera como una semana ideal de trabajo, donde los miembros de los equipos de desarrollo trabajan el tiempo planeado sin ningún tipo de interrupción. (11) Esta estimación incluye todo el esfuerzo asociado a la implementación de la historia de usuario, por ejemplo, las pruebas unitarias, la integración y refactorización del código, y la preparación y ejecución de las pruebas de aceptación.

Para el buen desarrollo del sistema propuesto, se realizó una estimación para cada una de las historias de usuario identificadas, llegando a los resultados que se muestran a continuación:

**Tabla 2. 24 Estimación de esfuerzos por HU**

Historia de Usuario	Puntos de Estimación
Autenticar	1
Gestionar documentación	1

Gestionar información de proyecto	1
Gestionar miembros por proyecto	1
Gestionar noticias	1
Gestionar Fórum	1
Gestionar Encuesta	1
Gestionar Enlaces	1
Imprimir documentos	1
Buscar información	1
Autenticación contra dominio LDAP	1
Gestionar cronograma de revisiones	1
Gestionar preguntas frecuentes	1
Mostrar el mapa de navegación	1
Gestionar solicitud de revisión	1
Insertar resultado de revisión	1
Mostrar reporte de resultado de revisión	1
Insertar resultado de auditoría	1
Mostrar reporte de resultado de auditorías	1
Gestionar eventos.	1
Mostrar cantidad de no conformidades a los jefes de proyectos.	1
Publicar cursos de formación	1

El tiempo total estimado para el desarrollo del portal es de 22 puntos de estimación, que equivalen a 5 meses y una semana ideales de trabajo interrumpibles.

### 2.7.2.2. Plan de iteraciones

Después de ser descritas e identificadas las historias de usuario y estimado el esfuerzo propuesto para la realización de cada una de ellas, se procede a la planificación de la etapa de implementación del sistema. Este plan especifica exactamente cuales historias de usuario serán implementadas para cada iteración del sistema y las posibles fechas para estas liberaciones. En base a lo antes mencionado se decide realizar el sistema en tres iteraciones, las cuales se detallan a continuación:

#### 2.7.2.2.1. Iteración 1

Esta iteración tiene como objetivo la implementación de las historias de usuario de mayor prioridad. Al finalizar se contará con las funcionalidades descritas en las historias de usuario 1, 2, 3, 11, 12,15, 16, 17, 18 y 19, las cuales hacen alusión a la autenticación del sistema como parte de la seguridad de la aplicación, gestionar los documentos, mostrar la información de los proyectos, autenticación contra dominio LDAP, mostrar el cronograma de revisiones, gestionar solicitud de revisión, insertar resultado de revisión, mostrar reporte de resultado de revisión, insertar el resultado de las auditorías y mostrar

resultado de auditoría . Además se tendrá la primera versión de prueba, la cual será mostrada al cliente con el objetivo de obtener una retroalimentación para el grupo de trabajo.

### 2.7.2.2.2. Iteración 2

El objetivo de esta iteración es la implementación de las restantes funcionalidades con prioridad media. Con la culminación de la misma se tendrán implementadas las peticiones del cliente descritas en las historias de usuario 4, 5, 6, 7 y 22, en las cuales se hace mención a la posibilidad de mostrar los miembros por proyectos de equipo de trabajo, gestionar noticias, gestionar un fórum de discusión sobre un tema relacionado con la Calidad del Software, gestionar una encuesta para medir la utilidad del sistema para los usuarios, los enlaces y publicar cursos de formación para los aseguradores de calidad de cada proyecto. La versión de prueba referente a esta iteración junto a las implementaciones anteriores, serán mostradas al cliente con el objetivo de realizar cambios en base a la opinión del mismo.

### 2.7.2.2.3. Iteración 3

En esta iteración serán implementadas las funcionalidades de prioridad baja. Estas funciones tienen el propósito de brindar al cliente comodidad en la gestión de otras tareas asociadas a las de alta prioridad, entre ellas se tienen: permitir la búsqueda de manera rápida de información en la aplicación, permitir la impresión de documentos, mostrar las preguntas frecuentes, el mapa de navegación del portal de calidad, gestionar eventos y mostrar la cantidad de no conformidades a los jefes de proyectos. Estas funciones están descritas en las historias de usuario 9, 10, 13, 14, 20 y 21 respectivamente. Como resultado de esta iteración se tendrá la versión 1.0 del producto final, adicionando lo concerniente a la interfaz web. A partir de este momento el sistema será puesto a prueba por un período de tiempo para evaluar el desempeño del mismo.

### 2.7.2.3. Plan de duración de las iteraciones

Como parte del ciclo de vida de un proyecto utilizando la metodología XP se crea el plan de duración de cada una de las iteraciones, en este caso se hace para el único equipo de desarrollo con el cual se cuenta. Este plan tiene como finalidad mostrar la duración de cada iteración, así como el orden en que serán implementadas las historias de usuario en cada una de las mismas.

**Tabla 2. 25 Plan de duración de las iteraciones**

<b>Iteraciones</b>	<b>Orden de las Historias de usuario a implementar</b>	<b>Duración total de las iteraciones</b>
--------------------	--	--

## Capítulo 2: Características del Sistema

Iteración 1	Autenticar Gestionar documentación Gestionar información de proyectos Autenticación contra dominio LDAP Gestionar cronograma de revisiones Gestionar solicitud de revisión Insertar resultado de revisión Mostrar reporte de resultado de revisión Insertar resultado de auditoría Mostrar reporte de resultado de auditoría	10 semanas
Iteración 2	Gestionar miembros por proyecto Gestionar Noticias Gestionar Fórum Gestionar Encuesta Gestionar Enlaces Publicar cursos de formación	6 semanas
Iteración 3	Buscar información Imprimir documentos Gestionar preguntas frecuentes Mostrar el mapa de navegación Gestionar eventos. Mostrar cantidad de no conformidades a los jefes de proyectos	6 semanas

### 2.7.2.4. Plan de entregas

En el plan de entrega que se plantea a continuación se hace una propuesta de la fecha aproximada en que se harán versiones (releases) al sistema al finalizar cada iteración en la fase de implementación.

**Tabla 2. 26 Plan de entrega**

Entregable	Final 1ra iteración 2da semana de marzo	Final 2da iteración 3ra semana de abril	Final 3ra iteración 1ra semana de junio
Portal de Calidad.	0.1	0.2	1.0

### 2.8. Conclusiones

En este capítulo se obtuvo un listado de funcionalidades que debe tener el sistema, expresados en los requerimientos funcionales. Partiendo de este punto, base de todo proceso de desarrollo, se puede comenzar con la construcción de la propuesta, velando por el cumplimiento de todos los requerimientos y funcionalidades consideradas. También se hace referencia a todo lo concerniente a la metodología seleccionada para construir del sistema, haciendo una descripción de cada uno de los artefactos

## Capítulo 2: Características del Sistema

---

generados en el transcurso del mismo como son las historias de usuario, el plan de iteraciones y el de entrega y se establece la estructura general que tendrá el portal de calidad.

# CAPÍTULO 3: CONSTRUCCIÓN DEL SISTEMA

## 3.1. Introducción

En el presente capítulo se describen las fases de diseño, implementación y prueba, propias de la metodología de desarrollo XP. Se detallan las 3 iteraciones realizadas durante la etapa de codificación del proyecto, así como las tareas de programación y pruebas de aceptación desarrolladas en cada una de estas.

## 3.2. Diseño

Con el objetivo de la comprensión de este trabajo se hace una breve explicación del funcionamiento de Drupal como plataforma de publicación.

Todo el contenido creado en Drupal es funcionalmente un objeto (nodo), lo cual permite que se le brinde un tratamiento uniforme a la información. Unido a este contenido creado existen enlaces permanentes (permalinks), con el objetivo de que pueda ser enlazado externamente sin temor a que este enlace falle en un futuro. Drupal posee además un sistema de control de versiones que permite auditar las sucesivas actualizaciones del contenido, o sea, que se ha cambiado, hora y fecha, quién lo cambió, etc.

Drupal posee un potente sistema de seguridad basado en roles, lo cual permite controlar el acceso a los diferentes módulos según los permisos que el administrador le haya definido a cada rol. (9)

### 3.2.1. Estructura de Drupal

La configuración típica de un sitio Drupal contiene 4 ficheros principales: Themes, Includes, Scripts y Modules. Debido a que cada uno de estos ficheros contiene a su vez gran cantidad de archivos se hará referencia a los mismos con el término paquete.

El paquete de Drupal constituye básicamente en la distribución del Sistema de Gestión de Contenido. Dentro del paquete Themes se puede encontrar el mecanismo de soporte de plantillas que permiten configurar el diseño de toda la capa de presentación del sistema; cuando se desee incluir un nuevo diseño, se copia la plantilla dentro de esta carpeta. El tema utilizado para la aplicación fue el Danger4k al cual se le realizaron modificaciones para adecuarlo a los requerimientos del cliente.

El paquete Includes contiene una serie de ficheros de configuración del sistema, los scripts para generar la base de datos, y clases utilitarias. Dentro del paquete Scripts se encuentran un conjunto de ficheros que se encargan de manejar el aspecto visual de la aplicación, fundamentalmente archivos CSS y java script. Por último dentro del paquete Modules se encuentran los diferentes módulos que brindan las diversas funcionalidades de Drupal, para incorporar un nuevo módulo sólo se tiene que copiar en el directorio donde se encuentran los mismos.

En el presente trabajo sólo se modelará el paquete Modules de Drupal, ya que este es el que contiene las principales funcionalidades del CMS.

Básicamente Drupal contiene una única página servidor, la cual genera el contenido a mostrar en la página final, de acuerdo a los argumentos con que se realicen la petición. Estas páginas generadas por Drupal pueden o no contener formularios, esto depende del módulo sobre el que se esté trabajando y el propósito del mismo. A continuación se representa el funcionamiento de Drupal como aplicación web.

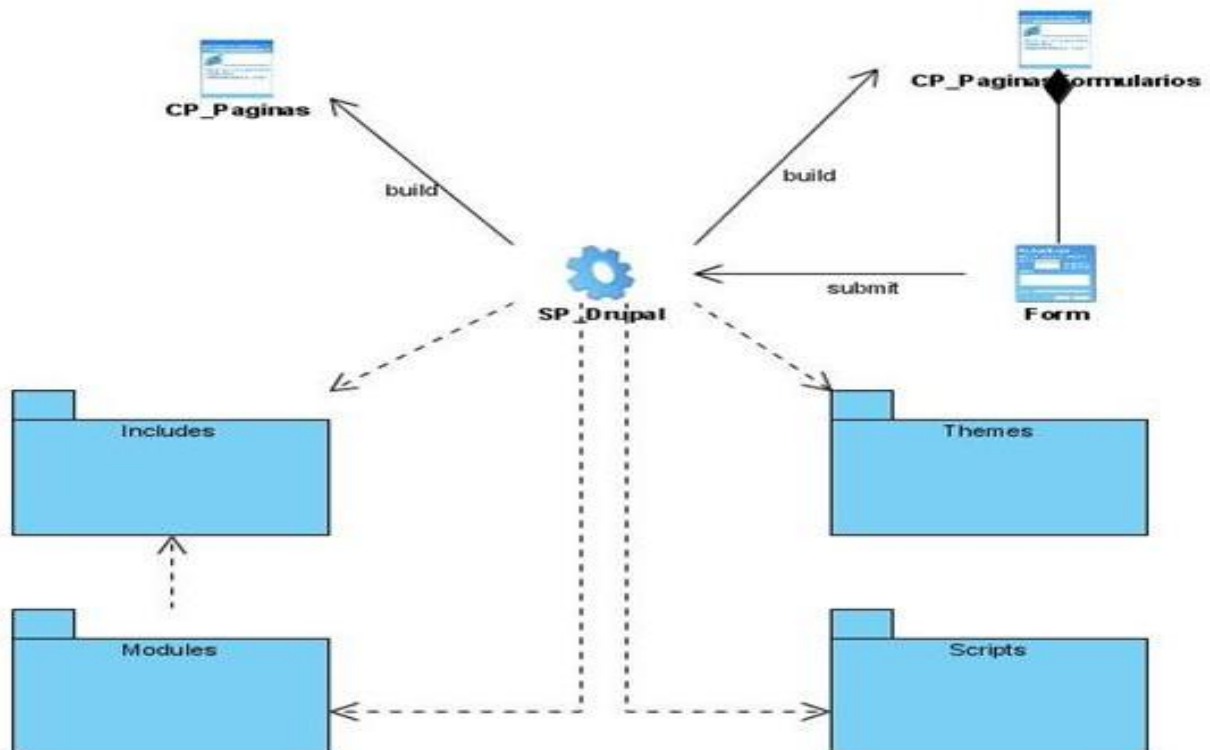


Figura 3 Diagrama de paquete de Drupal.



### 3.2.1.1. Módulos de Drupal

Dentro del paquete Modules se localizan dos grupos de módulos existentes en Drupal, los módulos que conforman el núcleo del CMS (Core Modules), es decir, los que trae por defecto en su instalación para lograr un perfecto funcionamiento; y los módulos contribuidos (Contrib Modules) que son plugins para extender, construir o modificar las funcionalidades del núcleo de Drupal.

Los módulos en Drupal son ficheros con extensión .module que contienen funciones escritas en PHP. Estas funciones actúan como funciones gancho (hooks), que son llamadas por Drupal durante sus procesos habituales de gestión de contenido y construcción de páginas web. (17) Por ejemplo, cada vez que un nodo es creado, visualizado, modificado o borrado, Drupal llama a uno de estos hooks pasándoles el contenido del nodo. De esta forma los módulos tienen la posibilidad de modificar y adaptar la información a visualizar en las páginas web antes de que se mande definitivamente al navegador.

La mayoría de los módulos en Drupal se instalan simplemente copiándolos en el directorio modules. Actualmente existe un número realmente grande de módulos disponibles. De temáticas diversas y de gran calidad. Antes de decidirse por un módulo u otro se deben revisar los requerimientos de los mismos.

Los módulos específicamente en Drupal permiten extender la funcionalidad de la web, añadiendo, por ejemplo, la posibilidad de realizar búsquedas por palabras dentro del contenido o de agregar comentarios a los nodos. En la página oficial de Drupal se puede encontrar un gran número de información detallada acerca de los procesos de construcción, instalación, configuración, actualización y desinstalación de módulos. (8)

### 3.2.1.2. Módulos del Portal de Calidad

Para el desarrollo del portal de Calidad se usó un gran número de módulos, estos serán divididos en dos grupos para lograr un mejor entendimiento de los mismos:

- 1. Módulos por defecto:** módulos que como su nombre lo indica, van a ser los que traía el Drupal o los que se encontraban en el portal oficial de la aplicación.
- 2. Módulos implementados:** módulos que fueron implementados para cumplir con una función específica del portal a desarrollar.

#### Módulos por defecto

- **CCK:** Para la creación de tipos de contenido personalizado.

- **Views (Vista):** Para obtención de vistas de los contenidos existentes.
- **LDAP\_Integration:** Para la autenticación de un dominio.
- **Node (Nodo):** Permite que el contenido que se presentará en el sitio se muestre en las páginas.
- **Upload (Cargar):** Subir y anexar archivos al contenido.
- **User:** Administra el registro de usuarios y sistema de acceso.
- **Search (Buscar):** Para realizar búsquedas de contenidos existentes.
- **Filter (Filtro):** Maneja el filtrado de contenido en preparación para su visualización.
- **System (Sistema):** Gestiona la configuración general del sitio.
- **Menú:** Para gestionar los diferentes menús.
- **Date (Fecha):** Permite el tratamiento de fecha y hora. Para agregar un campo de tipo fecha para la creación de contenidos.
- **Fórum:** Permite establecer debates sobre temas generales.
- **Calendar (Calendario):** Muestra cualquier tipo de campo de fecha y crea las vistas del calendario por defecto.
- **FCKeditor:** Habilita un editor de texto para campos de texto.
- **IMCE:** Habilita un soporte para anexar archivos e imágenes a los campos de texto a través del módulo FCKeditor.
- **Site map (Mapa del Sitio):** Muestra el mapa web del portal.
- **Print (Imprimir):** Permite generar una versión del contenido a pdf, impresión y envío por correo.
- **Block (Bloque):** Controla las casillas (bloques) que se muestran en el contenido principal.
- **Event (Evento):** Gestiona los eventos mediante una calendario.(8)

### Módulos implementados

- **Insertar:** Módulo para insertar el resultado de las revisiones.
- **Insertar auditoría:** Módulo para insertar el resultado de las auditorías.
- **Reportes:** Módulo para mostrar reportes por fecha y nombre de proyecto seleccionado, previamente revisado.
- **Reportes auditoría:** Módulo para mostrar reportes por fecha y nombre de proyecto seleccionado, previamente auditado.

### 3.2.2. Patrones de diseño en Drupal

Los patrones de diseño son aquellos que expresan esquemas para definir estructuras de diseño (o sus relaciones) con las que construir sistemas software.

El funcionamiento de Drupal hace uso de algunos patrones de diseño que son propiamente de sistemas orientados a objetos, particularmente los patrones GOF que contribuyen a la construcción de un diseño más elegante y robusto. Dentro de los patrones GOF existen varias categorías como son:

- **Creacionales** (abstraen el proceso de creación de instancias).
  - Singleton (Única instancia).

Si se piensan los módulos como objetos entonces siguiendo este patrón, se puede mencionar que en general estos objetos no encapsulan datos, lo que separa un módulo de otro es el conjunto de funciones que contiene, garantizando de esta manera la existencia de una única instancia de este objeto y la creación de un mecanismo de acceso global a dicha instancia, por lo que debe ser tratado como una clase con una instancia singleton.

- **Estructurales** (se ocupan de cómo clases y objetos son utilizados para componer estructuras de mayor tamaño).
  - Decorator (Envoltorio).

Añade funcionalidad a una clase dinámicamente.

EL uso de `hook_nodeapi`, permite a los módulos extender arbitrariamente el comportamiento de todos los nodos. Para los archivos cargados y adjuntados a un nodo, se podría diseñar un nuevo tipo de nodo con las demás características del nodo, que brinde la posibilidad de cargar archivos con la concesión de que cada nodo solicite la posibilidad de tener archivos adjuntos.

Este comportamiento podría ser imitado por el uso del patrón decorador envolviéndolo alrededor de cada objeto nodo.

- Bridge (Puente).

Los módulos deben ser escritos independientemente del sistema de base de datos que se está utilizando, y la capa de abstracción esta proporcionada para ello. Se pueden definir nuevas capas de base de datos de acuerdo a la API definida, añadiendo soporte para otros sistemas de bases de datos sin necesidad de modificar el código del módulo. Desacoplando de esta manera la abstracción de su implementación.

- **Comportamiento** (atañen a los algoritmos y a la asignación de responsabilidades entre objetos).
  - Observer (Observador).

Cuando se hace una modificación a un vocabulario del sistema de taxonomía de Drupal, el gancho taxonomía se llama en todos los módulos que lo implementan. Por la implementación del gancho, estos módulos quedan como observadores del objeto vocabulario; cualquier cambio que sufra este objeto, los módulos dependientes del objeto actúan tanto como sea apropiado.

- Command (Orden).

Muchos de los ganchos de Drupal utilizan el patrón de mando para reducir el número de funciones que son necesarias para la aplicación, pasando la operación como un parámetro, junto con los argumentos. De hecho, el propio sistema de gancho utiliza este patrón, a fin de que los módulos no tengan que definir cada gancho.

- Chain of Responsibility (Cadena de responsabilidad).

En cada solicitud de la página, el menú del sistema determina si hay un módulo para gestionar la solicitud, si el usuario tiene acceso a los recursos solicitados, y qué función se llama para hacer el trabajo. Para ello, el mensaje se pasa a la opción del menú, correspondiente a la vía de la solicitud. Si el elemento de menú no puede manejar la petición, se pasa a otro eslabón de la cadena. Esto continúa hasta que un módulo se encarga de la petición, un módulo niega el acceso para el usuario, o la cadena se ha agotado.

Por último, un patrón determinante es el “**Reflection Pattern**” (Patrón de reflexión), que describe el comportamiento de todo sitio basado en Drupal. Gracias a este patrón el sistema se convierte en una aplicación muy adaptable a diferentes entornos puesto que permite prácticamente la modificación de todo su comportamiento a través de los módulos instalables, sin la necesidad de modificar el núcleo.

### 3.3. Implementación

Durante el transcurso de las iteraciones se realiza la implementación de las historias de usuario seleccionadas para ser realizadas en cada una de ellas. Al principio de estas se lleva a cabo una revisión del plan de iteraciones y se modifica en caso de ser necesario. Como parte de este plan, se descomponen las HU en tareas de desarrollo, asignando a un grupo de desarrollo (o una persona), responsable de su implementación. Estas tareas son para el uso estricto de los programadores, pueden ser escritas en lenguaje técnico y no necesariamente entendible por el cliente.

Teniendo en cuenta la planificación realizada anteriormente, se llevaron a cabo tres iteraciones de desarrollo sobre el sistema, obteniéndose como finalidad un producto con todas las restricciones y características deseadas para ser utilizado. A continuación se detallan cada una de las iteraciones.

### 3.3.1. Iteración 1

En esta primera iteración se desarrollan las historias de usuarios de mayor prioridad en el sistema con el objetivo de obtener una primera versión del producto con las principales características o funcionalidades para ser mostrado al cliente.

**Tabla 3. 1 Historias abordadas en la primera iteración.**

No.	Historia de Usuario	Estimación	Real
1	Autenticar	1	1
2	Gestionar documentación	1	1
3	Gestionar información de proyectos	1	1
4	Autenticación contra dominio LDAP	1	1
5	Gestionar cronograma de revisiones	1	1
6	Gestionar solicitud de revisión	1	1
7	Insertar resultado de revisión	1	1
8	Mostrar reporte de resultado de revisión	1	1
9	Insertar resultado de auditoría	1	1
10	Mostrar reporte de resultado de auditoría	1	1
<b>Total</b>		<b>10</b>	<b>10</b>

#### 3.3.1.1. Tareas generadas por cada historia de usuario

##### HU Autenticar

**Tabla 3. 2 Tarea #1 HU Autenticar.**

Tarea	
Número de tarea: 1	Número de historia: 1
Nombre de la tarea: Autenticar	
Tipo de tarea: Configuración	Puntos estimados: 1
Fecha inicio: 2009-01-5	Fecha fin: 2009-01-9
Programador responsable: Anaisel Hernández Izquierdo	
Descripción: Se especificarán los diferentes datos que serán de utilidad para la creación, modificación y	

eliminación de la cuenta de usuario. Se especifica el rol al cual estará sujeta la cuenta para poder autenticarse el usuario.

### HU Gestionar documentación

**Tabla 3. 3 Tarea #1 HU Gestionar documentación.**

Tarea	
Número de tarea: 1	Número de historia: 2
Nombre de la tarea: Utilizar el tipo de contenido página.	
Tipo de tarea: Desarrollo	Puntos estimados: 1
Fecha inicio: 2009-01-12	Fecha fin: 2009-01-14
Programador responsable: Yanny Leyva Fajardo	
Descripción: Se utilizará el tipo de contenido página que trae consigo el CMS Drupal.	

**Tabla 3. 4 Tarea #2 HU Gestionar documentación.**

Tarea	
Número de tarea: 2	Número de historia: 2
Nombre de la tarea: Configurar opciones del menú de los contenidos tipo página.	
Tipo de tarea: Desarrollo	Puntos estimados: 1
Fecha inicio: 2009-01-15	Fecha fin: 2009-01-18
Programador responsable: Yanny Leyva Fajardo y Anaisel Hernández Izquierdo	
Descripción: Se configurarán los enlaces de las páginas para poder mostrar el contenido adjuntado, cuyo contenido es: las normas y estándares de calidad, los lineamientos de calidad, el expediente de proyecto, auditorías y revisiones, criterios de criticidad y los casos de estudio del expediente de proyecto.	

### HU Autenticación contra dominio LDAP

**Tabla 3. 5 Tarea #1 HU Autenticación contra dominio LDAP.**

Tarea	
Número de tarea: 1	Número de historia: 11
Nombre de la tarea: Agregar el módulo LDAP_Integration.	
Tipo de tarea: Configuración	Puntos estimados: 1
Fecha inicio: 2009-01-19	Fecha fin: 2009-01-22
Programador responsable: Anaisel Hernández Izquierdo	
Descripción: Se agregará el módulo LDAP_Integration en la carpeta de módulos del CMS y se activará el	

sub\_módulo ldapauth. De esta manera los usuarios se registrarán por el dominio UCI.

### HU Gestionar Información de proyecto

**Tabla 3. 6 Tarea #1 HU Gestionar Información de proyecto.**

Tarea	
<b>Número de tarea:</b> 1	<b>Número de historia:</b> 3
<b>Nombre de la tarea:</b> Crear el tipo de contenido Información_Proyecto.	
<b>Tipo de tarea:</b> Desarrollo	<b>Puntos estimados:</b> 1
<b>Fecha inicio:</b> 2009-01-26	<b>Fecha fin:</b> 2009-01-26
<b>Programador responsable:</b> Yanny Leyva Fajardo	
<b>Descripción:</b> Se creará el tipo de contenido Información_Proyecto para publicar de esta forma el expediente de proyecto. Deberá estar activado previamente el módulo CCK.	

**Tabla 3. 7 Tarea #2 HU Gestionar Información de proyecto.**

Tarea	
<b>Número de tarea:</b> 2	<b>Número de historia:</b> 3
<b>Nombre de la tarea:</b> Añadir los campos del tipo de contenido Información_Proyecto.	
<b>Tipo de tarea:</b> Desarrollo	<b>Puntos estimados:</b> 1
<b>Fecha inicio:</b> 2009-01-27	<b>Fecha fin:</b> 2009-01-28
<b>Programador responsable:</b> Yanny Leyva Fajardo	
<b>Descripción:</b> Se añadirán los campos necesarios del tipo de contenido Información_Proyecto para poder insertar los datos específicos de la misma. Deberá estar activado previamente el módulo CCK.	

**Tabla 3. 8 Tarea #3 HU Gestionar Información de proyecto.**

Tarea	
<b>Número de tarea:</b> 3	<b>Número de historia:</b> 3
<b>Nombre de la tarea:</b> Crear dos vistas para la información de proyecto.	
<b>Tipo de tarea:</b> Desarrollo	<b>Puntos estimados:</b> 1
<b>Fecha inicio:</b> 2009-01-29	<b>Fecha fin:</b> 2009-01-29
<b>Programador responsable:</b> Yanny Leyva Fajardo	
<b>Descripción:</b> Se crearán dos vistas para la información de proyecto y se procederá a configurar las mismas. Deberá estar activado previamente el módulo Views.	

**Tabla 3. 9 Tarea #4 HU Gestionar Información de proyecto.**

Tarea	
Número de tarea: 4	Número de historia:3
Nombre de la tarea: Configurar las vistas de la información de proyecto.	
Tipo de tarea: Desarrollo	Puntos estimados: 1
Fecha inicio: 2009-01-30	Fecha fin: 2009-01-31
Programador responsable: Yanny Leyva Fajardo	
<b>Descripción:</b> Se configurará la vista de la información de proyecto a través de las siguientes opciones: <ul style="list-style-type: none"> <li>• Información básica.</li> <li>• Uso de página.</li> <li>• Filtro (aquí se define el tipo de noticia a mostrar).</li> <li>• Campos a mostrar.</li> </ul>	

### HU Gestionar Cronograma de revisiones

**Tabla 3. 10 Tarea #1 HU Gestionar cronograma de revisiones.**

Tarea	
Número de tarea: 1	Número de historia: 12
Nombre de la tarea: Crear el tipo de contenido cronograma de revisiones.	
Tipo de tarea: Desarrollo	Puntos estimados: 1
Fecha inicio: 2009-02-2	Fecha fin: 2009-02-2
Programador responsable: Anaisel Hernández Izquierdo	
<b>Descripción:</b> Se creará el tipo de contenido cronograma de revisiones para publicar de esta forma el cronograma de revisiones. Deberá estar activado previamente el módulo CCK.	

**Tabla 3. 11 Tarea #2 HU Gestionar cronograma de revisiones.**

Tarea	
Número de tarea: 2	Número de historia: 12
Nombre de la tarea: Añadir los campos del tipo de contenido cronograma de revisiones.	
Tipo de tarea: Desarrollo	Puntos estimados: 1
Fecha inicio: 2009-02-3	Fecha fin: 2009-02-3
Programador responsable: Anaisel Hernández Izquierdo	
<b>Descripción:</b> Se añadirán los campos necesarios del tipo de contenido cronograma de revisiones para poder	



insertar los datos específicos de la misma. Deberá estar activado previamente el módulo CCK.

**Tabla 3. 12 Tarea #3 HU Gestionar cronograma de revisiones.**

Tarea	
<b>Número de tarea:</b> 3	<b>Número de historia:</b> 12
<b>Nombre de la tarea:</b> Crear una vista para el cronograma de revisiones.	
<b>Tipo de tarea:</b> Desarrollo	<b>Puntos estimados:</b> 1
<b>Fecha inicio:</b> 2009-02-4	<b>Fecha fin:</b> 2009-02-5
<b>Programador responsable:</b> Yanny Leyva Fajardo	
<b>Descripción:</b> Se creará una vista para el cronograma de revisiones y se procederá a configurar la misma. Deberá estar activado previamente el módulo Views.	

**Tabla 3. 13 Tarea #4 HU Gestionar cronograma de revisiones.**

Tarea	
<b>Número de tarea:</b> 4	<b>Número de historia:</b> 12
<b>Nombre de la tarea:</b> Configurar la vista del cronograma.	
<b>Tipo de tarea:</b> Desarrollo	<b>Puntos estimados:</b> 1
<b>Fecha inicio:</b> 2009-02-6	<b>Fecha fin:</b> 2009-02-8
<b>Programador responsable:</b> Yanny Leyva Fajardo	
<b>Descripción:</b> Se configurará la vista del cronograma a través de las siguientes opciones: <ul style="list-style-type: none"> <li>• Información básica.</li> <li>• Uso de página.</li> <li>• Filtro (aquí se define el tipo de noticia a mostrar).</li> <li>• Campos a mostrar.</li> </ul>	

### HU Gestionar solicitud de revisión

**Tabla 3. 14 Tarea #1 HU Gestionar solicitud de revisión.**

Tarea	
<b>Número de tarea:</b> 1	<b>Número de historia:</b> 15
<b>Nombre de la tarea:</b> Crear el tipo de contenido solicitud de revisión.	
<b>Tipo de tarea:</b> Desarrollo	<b>Puntos estimados:</b> 1
<b>Fecha inicio:</b> 2009-02-9	<b>Fecha fin:</b> 2009-02-9

<b>Programador responsable:</b> Yanny Leyva Fajardo.
<b>Descripción:</b> Se creará el tipo de contenido solicitud de revisión para poder de esta forma poder enviar y recibir una solicitud de revisión. Deberá estar activado previamente el módulo CCK.

**Tabla 3. 15 Tarea #2 HU Gestionar solicitud de revisión.**

Tarea	
<b>Número de tarea:</b> 2	<b>Número de historia:</b> 15
<b>Nombre de la tarea:</b> Añadir los campos del tipo de contenido solicitud de revisión.	
<b>Tipo de tarea:</b> Desarrollo	<b>Puntos estimados:</b> 1
<b>Fecha inicio:</b> 2009-02-10	<b>Fecha fin:</b> 2009-02-10
<b>Programador responsable:</b> Yanny Leyva Fajardo	
<b>Descripción:</b> Se añadirán los campos necesarios del tipo de contenido solicitud de revisión para poder insertar los datos específicos de la misma. Deberá estar activado previamente el módulo CCK.	

**Tabla 3. 16 Tarea #3 HU Gestionar solicitud de revisión.**

Tarea	
<b>Número de tarea:</b> 3	<b>Número de historia:</b> 15
<b>Nombre de la tarea:</b> Crear una vista para el tipo de contenido solicitud de revisión.	
<b>Tipo de tarea:</b> Desarrollo	<b>Puntos estimados:</b> 1
<b>Fecha inicio:</b> 2009-02-11	<b>Fecha fin:</b> 2009-02-12
<b>Programador responsable:</b> Yanny Leyva Fajardo	
<b>Descripción:</b> Se creará una vista para solicitud de revisión y se procederá a configurar la misma. Deberá estar activado previamente el módulo Views.	

**Tabla 3. 17 Tarea #4 HU Gestionar solicitud de revisión.**

Tarea	
<b>Número de tarea:</b> 4	<b>Número de historia:</b> 15
<b>Nombre de la tarea:</b> Configurar la vista solicitud_rev.	
<b>Tipo de tarea:</b> Desarrollo	<b>Puntos estimados:</b> 1
<b>Fecha inicio:</b> 2009-02-12	<b>Fecha fin:</b> 2009-02-15
<b>Programador responsable:</b> Yanny Leyva Fajardo	
<b>Descripción:</b> Se configurará la vista solicitud_rev a través de las siguientes opciones:	

- Información básica.
- Uso de página.
- Filtro (aquí se define el tipo de noticia a mostrar).
- Campos a mostrar.

### HU Insertar resultado de revisión

Tabla 3. 18 Tarea #1 HU Insertar resultado de revisión

Tarea	
Número de tarea: 1	Número de historia: 16
Nombre de la tarea: Activar el módulo desarrollado Insertar.	
Tipo de tarea: Desarrollo	Puntos estimados: 1
Fecha inicio: 2009-02-16	Fecha fin: 2009-02-22
Programador responsable: Anaisel Hernández Izquierdo	
Descripción: Se activará el módulo desarrollado Insertar para de esta forma poder insertar los datos del resultado de la revisión.	

### HU Mostrar reporte de resultado de revisión

Tabla 3. 19 Tarea #1 HU Mostrar reporte de resultado de revisión.

Tarea	
Número de tarea: 1	Número de historia: 17
Nombre de la tarea: Activar el módulo desarrollado Reportes.	
Tipo de tarea: Desarrollo	Puntos estimados: 1
Fecha inicio: 2009-02-23	Fecha fin: 2009-02-28
Programador responsable: Anaisel Hernández Izquierdo	
Descripción: Se activará el módulo desarrollado reportes para de esta forma poder sacar reportes ya sea por la fecha seleccionada o por el nombre del proyecto después de haber insertado el resultado de la revisión.	

### HU Insertar resultado de auditoría

Tabla 3. 20 Tarea #1 HU Insertar resultado de auditoría.

Tarea	
Número de tarea: 1	Número de historia: 18
Nombre de la tarea: Activar el módulo desarrollado Insertar_auditoría.	

<b>Tipo de tarea:</b> Desarrollo	<b>Puntos estimados:</b> 1
<b>Fecha inicio:</b> 2009-03-2	<b>Fecha fin:</b> 2009-03-8
<b>Programador responsable:</b> Anaisel Hernández Izquierdo	
<b>Descripción:</b> Se activará el módulo desarrollado Insertar_ auditoría para de esta forma poder insertar los datos del resultado de la auditoría.	

**Tabla 3. 21 Tarea #1 HU Mostrar resultado de auditoría.**

Tarea	
<b>Número de tarea:</b> 1	<b>Número de historia:</b> 19
<b>Nombre de la tarea:</b> Activar el módulo desarrollado Reportes_auditoría.	
<b>Tipo de tarea:</b> Desarrollo	<b>Puntos estimados:</b> 1
<b>Fecha inicio:</b> 2009-03-9	<b>Fecha fin:</b> 2009-03-15
<b>Programador responsable:</b> Anaisel Hernández Izquierdo	
<b>Descripción:</b> Se activará el módulo desarrollado reportes_ auditoría para de esta forma poder sacar reportes ya sea por la fecha seleccionada o por el nombre del proyecto después de haber insertado el resultado de la auditoría.	

### 3.3.2. Iteración 2

En el transcurso de esta iteración se implementan las historias de usuarios con nivel de prioridad media para el sistema .Esto permite agregar nuevas funcionalidades al sistema por lo que se obtiene una versión más completa y el cliente podrá observar algunas de las facilidades en cuanto a las funcionalidades del sistema en su culminación.

**Tabla 3. 22 Historias abordadas en la segunda iteración.**

No.	Historia de Usuario	Estimación	Real
1	Gestionar miembros por proyecto.	1	1
2	Gestionar Noticias.	1	1
3	Gestionar Fórum.	1	1
4	Gestionar Encuesta.	1	1
5	Gestionar Enlaces.	1	1
6	Publicar cursos de formación	1	1
<b>Total</b>		<b>6</b>	<b>6</b>

### 3.3.2.1. Tareas generadas por cada historia de usuario

#### HU Gestionar miembros por proyecto

**Tabla 3. 23 Tarea #1 HU Gestionar miembros por proyecto.**

Tarea	
Número de tarea: 1	Número de historia: 4
Nombre de la tarea: Crear el tipo de contenido grupo.	
Tipo de tarea: Desarrollo	Puntos estimados: 1
Fecha inicio: 2009-03-16	Fecha fin: 2009-03-16
Programador responsable: Anaisel Hernández Izquierdo	
Descripción: Se creará el tipo de contenido grupo para insertar de esta forma los aseguradores de calidad y el nombre de los proyectos al que pertenece. Deberá estar activado previamente el módulo CCK.	

**Tabla 3. 24 Tarea #2 HU Gestionar miembros por proyecto.**

Tarea	
Número de tarea: 2	Número de historia: 4
Nombre de la tarea: Añadir los campos del tipo de contenido grupo.	
Tipo de tarea: Desarrollo	Puntos estimados: 1
Fecha inicio: 2009-03-17	Fecha fin: 2009-03-17
Programador responsable: Yanny Leyva Fajardo	
Descripción: Se añadirán los campos necesarios del tipo de contenido grupo para poder insertar los datos específicos de la misma. Deberá estar activado previamente el módulo CCK.	

**Tabla 3. 25 Tarea #3 HU Gestionar miembros por proyecto.**

Tarea	
Número de tarea: 3	Número de historia: 4
Nombre de la tarea: Crear una vista para el grupo.	
Tipo de tarea: Desarrollo	Puntos estimados: 1
Fecha inicio: 2009-03-18	Fecha fin: 2009-03-19
Programador responsable: Yanny Leyva Fajardo y Anaisel Hernández Izquierdo	
Descripción: Se crearán una vista para el grupo y se procederá a configurar la misma. Deberá estar activado previamente el módulo Views.	

**Tabla 3. 26 Tarea #4 HU Gestionar miembros por proyecto.**

Tarea	
Número de tarea: 4	Número de historia: 4
Nombre de la tarea: Configurar la vista del grupo.	
Tipo de tarea: Desarrollo	Puntos estimados: 1
Fecha inicio: 2009-03-20	Fecha fin: 2009-03-22
Programador responsable: Yanny Leyva Fajardo y Anaisel Hernández Izquierdo	
<p><b>Descripción:</b> Se configurará la vista del grupo a través de las siguientes opciones:</p> <ul style="list-style-type: none"> <li>• Información básica.</li> <li>• Uso de página.</li> <li>• Filtro (aquí se define el tipo de noticia a mostrar).</li> <li>• Campos a mostrar.</li> </ul>	

### HU Gestionar noticias

**Tabla 3. 27 Tarea #1 HU Gestionar noticias.**

Tarea	
Número de tarea: 1	Número de historia: 5
Nombre de la tarea: Crear el tipo de contenido noticia.	
Tipo de tarea: Desarrollo	Puntos estimados: 1
Fecha inicio: 2009-03-23	Fecha fin: 2009-03-23
Programador responsable: Yanny Leyva Fajardo	
<p><b>Descripción:</b> Se creará el tipo de contenido noticia para publicar de esta forma las noticias sobre calidad y actualidad informática. Deberá estar activado previamente el módulo CCK.</p>	

**Tabla 3. 28 Tarea #2 HU Gestionar noticias.**

Tarea	
Número de tarea: 2	Número de historia: 5
Nombre de la tarea: Añadir los campos del tipo de contenido noticia.	
Tipo de tarea: Desarrollo	Puntos estimados: 1
Fecha inicio: 2009-03-24	Fecha fin: 2009-03-24
Programador responsable: Yanny Leyva Fajardo	
<p><b>Descripción:</b> Se añadirán los campos necesarios del tipo de contenido noticia para poder insertar los datos</p>	

específicos de la misma. Deberá estar activado previamente el módulo CCK.

**Tabla 3. 29 Tarea #3 HU Gestionar noticias.**

Tarea	
<b>Número de tarea:</b> 3	<b>Número de historia:</b> 5
<b>Nombre de la tarea:</b> Crear dos vistas para las noticias.	
<b>Tipo de tarea:</b> Desarrollo	<b>Puntos estimados:</b> 1
<b>Fecha inicio:</b> 2009-03-25	<b>Fecha fin:</b> 2009-03-25
<b>Programador responsable:</b> Yanny Leyva Fajardo y Anaisel Hernández Izquierdo	
<b>Descripción:</b> Se crearán dos vistas para las noticias y se procederá a configurar las mismas. Deberá estar activado previamente el módulo Views.	

**Tabla 3. 30 Tarea #4 HU Gestionar noticias.**

Tarea	
<b>Número de tarea:</b> 4	<b>Número de historia:</b> 5
<b>Nombre de la tarea:</b> Configurar las vistas de las noticias.	
<b>Tipo de tarea:</b> Desarrollo	<b>Puntos estimados:</b> 1
<b>Fecha inicio:</b> 2009-03-26	<b>Fecha fin:</b> 2009-03-28
<b>Programador responsable:</b> Yanny Leyva Fajardo y Anaisel Hernández Izquierdo	
<b>Descripción:</b> Se configurará la vista de las noticias de calidad y la vista de las noticias de actualidad informática a través de las siguientes opciones: <ul style="list-style-type: none"> <li>• Información básica.</li> <li>• Uso de página.</li> <li>• Filtro (aquí se define el tipo de noticia a mostrar).</li> <li>• Campos a mostrar.</li> </ul>	

### HU Gestionar Fórum

**Tabla 3.31 Tarea #1 HU Gestionar Fórum.**

Tarea	
<b>Número de tarea:</b> 1	<b>Número de historia:</b> 6
<b>Nombre de la tarea:</b> Activar módulo Fórum.	
<b>Tipo de tarea:</b> Configuración	<b>Puntos estimados:</b> 1
<b>Fecha inicio:</b> 2009-03-30	<b>Fecha fin:</b> 2009-04-04

<b>Programador responsable:</b> Anaisel Hernández Izquierdo
<b>Descripción:</b> Se activará el módulo Fórum para permitir debates sobre el tema en cuestión en el sistema.

### HU Gestionar encuesta

**Tabla 3.32 Tarea #1 HU Gestionar encuesta.**

Tarea	
<b>Número de tarea:</b> 1	<b>Número de historia:</b> 7
<b>Nombre de la tarea:</b> Activar módulo Encuesta.	
<b>Tipo de tarea:</b> Configuración	<b>Puntos estimados:</b> 1
<b>Fecha inicio:</b> 2009-04-06	<b>Fecha fin:</b> 2009-04-11
<b>Programador responsable:</b> Yanny Leyva Fajardo	
<b>Descripción:</b> Se activará el módulo Encuesta para permitir obtener un resultado sobre el tema en cuestión en el sistema.	

### HU Gestionar enlaces

**Tabla 3. 33 Tarea #1 HU Gestionar enlaces.**

Tarea	
<b>Número de tarea:</b> 1	<b>Número de historia:</b> 8
<b>Nombre de la tarea:</b> Crear menú.	
<b>Tipo de tarea:</b> Desarrollo	<b>Puntos estimados:</b> 1
<b>Fecha inicio:</b> 2009-04-13	<b>Fecha fin:</b> 2009-04-18
<b>Programador responsable:</b> Anaisel Hernández Izquierdo	
<b>Descripción:</b> Se creará un menú donde se le adiciona un enlace por cada sitio a referenciar, configurando las opciones de cada uno.	

### HU Publicar cursos de formación.

**Tabla 3. 34 Tarea #1 HU Publicar cursos de formación.**

Tarea	
<b>Número de tarea:</b> 1	<b>Número de historia:</b> 22
<b>Nombre de la tarea:</b> Crear el tipo de contenido cursos y añadir los campos del mismo.	
<b>Tipo de tarea:</b> Desarrollo	<b>Puntos estimados:</b> 1
<b>Fecha inicio:</b> 2009-04-20	<b>Fecha fin:</b> 2009-04-21
<b>Programador responsable:</b> Yanny Leyva Fajardo	
<b>Descripción:</b> Se creará el tipo de contenido cursos y se añadirán los campos necesarios para poder insertar los datos específicos de los mismos. Deberá estar activado previamente el módulo CCK.	



**Tabla 3.35 Tarea #2 HU Publicar cursos de formación.**

Tarea	
Número de tarea: 2	Número de historia: 22
Nombre de la tarea: Crear una vista y configurarla para el tipo de contenido cursos.	
Tipo de tarea: Desarrollo	Puntos estimados: 1
Fecha inicio: 2009-04-23	Fecha fin: 2009-04-25
Programador responsable: Yanny Leyva Fajardo	
<p><b>Descripción:</b> Se creará y configurará la vista de los cursos a través de las siguientes opciones:</p> <ul style="list-style-type: none"> <li>Información básica.</li> <li>Uso de página.</li> <li>Filtro (aquí se define el tipo de noticia a mostrar).</li> </ul> <p>Campos a mostrar.</p> <p>Deberá estar activado previamente el módulo Views.</p>	

### 3.3.3. Iteración 3

Al finalizar esta iteración se habrán implementado todos los elementos y funcionalidades del sistema brindando una versión final del producto listo para ponerlo en funcionamiento.

**Tabla 3. 36 Historias abordadas en la tercera iteración.**

No.	Historia de Usuario	Estimación	Real
1	Buscar información.	1	1
2	Imprimir documentos.	1	1
3	Gestionar preguntas frecuentes	1	1
4	Mostrar el mapa de navegación	1	1
5	Gestionar eventos.	1	1
6	Mostrar cantidad de no conformidades a jefes de proyectos	1	1
<b>Total</b>		<b>6</b>	<b>6</b>

#### 3.3.3.1. Tareas generadas por cada historia de usuario

##### HU Buscar información

**Tabla 3. 37 Tarea #1 HU Buscar información.**

Tarea	
Número de tarea: 1	Número de historia: 9
Nombre de la tarea: Activar módulo Search.	

<b>Tipo de tarea:</b> Configuración	<b>Puntos estimados:</b> 1
<b>Fecha inicio:</b> 2009-04-27	<b>Fecha fin:</b> 2009-04-30
<b>Programador responsable:</b> Anaisel Hernández Izquierdo	
<b>Descripción:</b> Se activará el módulo Search para permitir la búsqueda de la información existente en el sistema. Previamente deberá estar activado el módulo poormanscron.	

### HU Imprimir información

**Tabla 3. 38 Tarea #1 HU Imprimir información.**

Tarea	
<b>Número de tarea:</b> 1	<b>Número de historia:</b> 10
<b>Nombre de la tarea:</b> Activar módulo Print.	
<b>Tipo de tarea:</b> Configuración	<b>Puntos estimados:</b> 1
<b>Fecha inicio:</b> 2009-05-04	<b>Fecha fin:</b> 2009-05-08
<b>Programador responsable:</b> Anaisel Hernández Izquierdo	
<b>Descripción:</b> Se activará el módulo Print para permitir la impresión de la información existente en el sistema.	

### HU Gestionar preguntas frecuentes

**Tabla 3. 39 Tarea #1 HU Gestionar preguntas frecuentes.**

Tarea	
<b>Número de tarea:</b> 1	<b>Número de historia:</b> 13
<b>Nombre de la tarea:</b> Utilizar el tipo de contenido página.	
<b>Tipo de tarea:</b> Desarrollo	<b>Puntos estimados:</b> 1
<b>Fecha inicio:</b> 2009-05-11	<b>Fecha fin:</b> 2009-05-15
<b>Programador responsable:</b> Yanny Leyva Fajardo	
<b>Descripción:</b> Se utilizará el tipo de contenido página que trae consigo el CMS Drupal. Se muestra el contenido.	

### HU Mostrar el mapa de navegación

**Tabla 3.40 Tarea #1 HU Mostrar el mapa de navegación.**

Tarea	
<b>Número de tarea:</b> 1	<b>Número de historia:</b> 14
<b>Nombre de la tarea:</b> Activar módulo Site Map.	
<b>Tipo de tarea:</b> Configuración	<b>Puntos estimados:</b> 1
<b>Fecha inicio:</b> 2009-05-18	<b>Fecha fin:</b> 2009-05-22

<b>Programador responsable:</b> Yanny Leyva Fajardo
<b>Descripción:</b> Se activará el módulo Site Map y se configurarán las opciones del mismo para brindarle al usuario el mapa de navegación del portal.

### HU Gestionar eventos

**Tabla 3. 41 Tarea #1 HU Gestionar eventos.**

Tarea	
<b>Número de tarea:</b> 1	<b>Número de historia:</b> 20
<b>Nombre de la tarea:</b> Activar módulo Event.	
<b>Tipo de tarea:</b> Configuración	<b>Puntos estimados:</b> 1
<b>Fecha inicio:</b> 2009-05-25	<b>Fecha fin:</b> 2009-05-23
<b>Programador responsable:</b> Yanny Leyva Fajardo	
<b>Descripción:</b> Se activará el módulo Event y se configurarán las opciones del mismo para brindarle al usuario los eventos necesarios para mantenerse informado.	

### HU Mostrar cantidad de no conformidades a jefes de proyectos

**Tabla 3. 42 Tarea #1 HU Mostrar cantidad de no conformidades a jefes de proyectos**

Tarea	
<b>Número de tarea:</b> 1	<b>Número de historia:</b> 21
<b>Nombre de la tarea:</b> Utilizar el tipo de contenido página.	
<b>Tipo de tarea:</b> Desarrollo	<b>Puntos estimados:</b> 1
<b>Fecha inicio:</b> 2009-06-1	<b>Fecha fin:</b> 2009-06-5
<b>Programador responsable:</b> Yanny Leyva Fajardo	
<b>Descripción:</b> Se utilizará el tipo de contenido página que trae consigo el CMS Drupal, al cual de le insertará un formulario a través de un código PHP para obtener un reporte de la cantidad de no conformidades insertadas anteriormente en las revisiones, y lo verá cada uno de las jefe de proyectos al entrar al sistema.	

#### 3.3.4. Desarrollo de los módulos en Drupal

Para crear un módulo se debe tener en cuenta que deben ser creados tres archivos PHP y ser guardados con el nombre del módulo seguido de .info que permite ofrecer información acerca del módulo, otro con el nombre del módulo .install para crear las tablas en la base de datos y por último un archivo con el nombre del módulo .module donde se escriben funciones que actúan en forma de enganche al ser llamadas por Drupal a la hora de construir una página web y gestionar el contenido.

Descripción del archivo .module

```
<?php  
/* $Id$ */
```

El código mostrado previamente es lo primero que debe ser incluido en este archivo donde la cadena \$Id\$ permite que el controlador de versiones de Drupal le asigne un valor distinto al módulo y una fecha de modificación cada vez que el mismo sea modificado posibilitando tener un control sobre los cambios que le son realizados al código.

Descripción del archivo.info

```
; $Id$  
name = Nombre del módulo  
description = "Una descripción de lo que hace el módulo."  
versión = "Módulos de los que depende."
```

El sistema de módulos de Drupal está basado en el concepto de "hooks". Un hook es una función de PHP a la que se llama foo\_bar (), donde "foo" es el nombre del módulo (cuyo nombre de archivo es también foo.module) y "bar" es el nombre del hook. Cada hook tiene una serie de parámetros predefinidos y el tipo de variable que devuelve.

Hook utilizados en el desarrollo de los módulos insertar, insertar auditoría, reportes y reportes auditoría.

**Hook\_access:** Define restricciones de acceso. Este hook permite a módulos de nodo limitar el acceso. La cuenta administrativa (el usuario ID #1) siempre pasa cualquier comprobación de acceso, no se llama a este hook en este caso. Si este hook no está definido para un tipo de nodo, todas las comprobaciones de acceso fallarán, tan sólo el administrador será capaz de ver el contenido.

**Hook\_perm:** Aquí es donde se definen los nombres de los permisos del módulo. Esta función no asigna los permisos, simplemente especifica cuáles estarán disponibles para el módulo. Con esta función se da acceso a quien puede acceder al contenido del sitio o administrar el módulo.

**Hook\_menu:** Es llamada siempre por Drupal antes de crear las páginas, antes de mandarlas al navegador, y en ella pueden indicarse los puntos concretos en los que el módulo interactuará con la web.

**Hook\_view:** Esto es un gancho usado por módulos de nodo. Esto permite a un módulo definir un método de mostrar los nodos.

**Hook\_form:** Muestra un nodo para editar los formularios que contenga. Este hook es usado por módulos de nodo, es llamado cuando se necesita crear o editar algún artículo. Este hook tiene que devolver el título de nodo, el área de texto, y otro campo específico del tipo de nodo. Un arreglo devuelve los elementos del formulario del nodo que se está modificando.

**Hook\_submit:** Es utilizado por módulos de nodo. Es usado para las acciones que deben ocurrir cuando el módulo es guardado. Es utilizado después de que la validación ha tenido éxito y antes de insertar o actualizar.

**Hook\_validate:** Esto es un gancho usado por módulos de nodo. Es utilizado para permitir al módulo verificar que el nodo se encuentra en un formato válido para añadirse al sitio.

**Hook\_delete:** Es un gancho usado por módulos de nodo. Permite al módulo tomar medidas cuando un nodo es eliminado de la base de datos, por ejemplo, borrando la información de tablas relacionadas.

### 3.3.5. Diagrama de despliegue

El diagrama de despliegue permite apreciar de forma visual cómo se encuentran relacionados físicamente los componentes de la aplicación. En este caso la aplicación se encuentra hospedada en un servidor web y la misma se comunica con un sistema de gestión de base de datos (PostgreSQL).

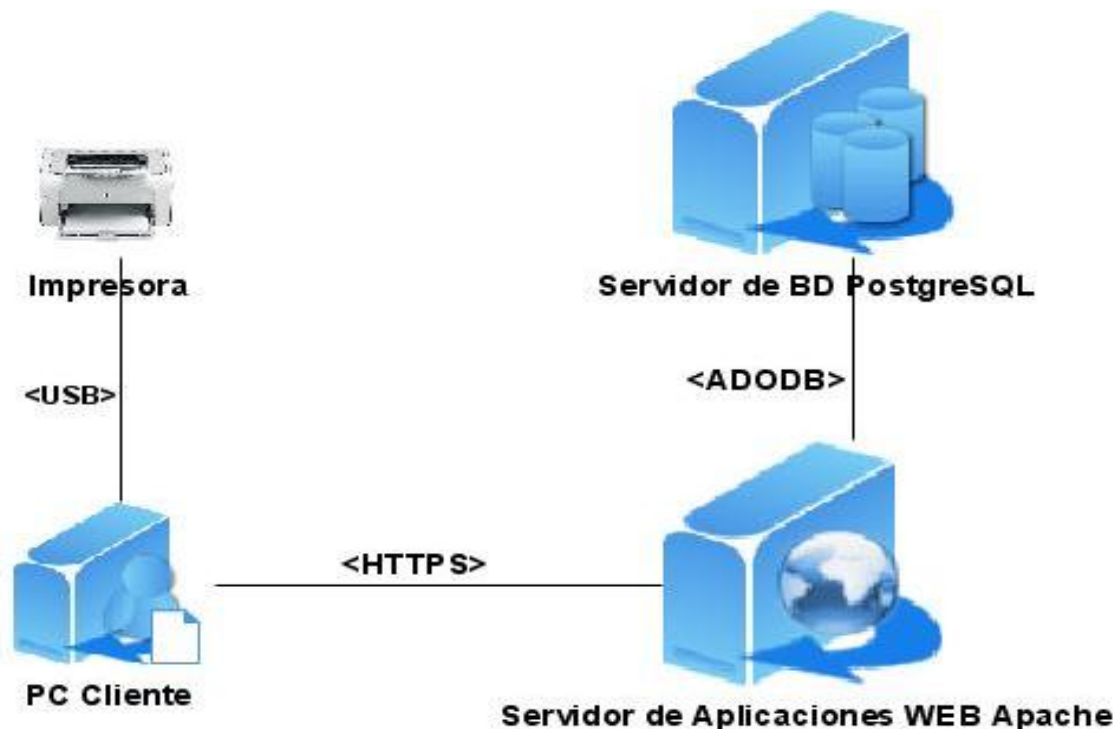


Figura 4 Diagrama de despliegue.

### 3.4. Pruebas

Uno de los pilares fundamentales de XP es el proceso de pruebas, el cual anima a los desarrolladores a probar constantemente tanto como sea posible. Mediante esta filosofía se reduce el número de errores no detectados así como el tiempo entre la introducción de este en el sistema y su detección. Todo esto contribuye a elevar la calidad de los productos desarrollados y a la seguridad de los programadores a la hora de introducir cambios o modificaciones.

La metodología XP divide las pruebas en dos grupos: pruebas unitarias, desarrolladas por los programadores, encargadas de verificar el código de forma automática y las pruebas de aceptación, destinadas a evaluar si al final de una iteración se obtuvo la funcionalidad requerida, además de comprobar que dicha funcionalidad sea la esperada por el cliente. (14)

#### 3.4.1. Pruebas unitarias

Dentro de las pruebas se encuentran dos fundamentales: Prueba de Caja Negra y de Caja Blanca. La prueba de la caja blanca del software se comprueba los caminos lógicos del software proponiendo casos

de prueba que se ejerciten conjuntos específicos de condiciones. Se puede examinar el estado del programa en varios puntos para determinar si el estado real coincide con el esperado o mencionado.

La prueba de caja negra se refiere a las pruebas que se llevan a cabo sobre la interfaz del software. O sea, a través de los casos de prueba se demuestra que las funciones del software son operativas, que la entrada se acepta de forma adecuada y que se produce un resultado correcto, así como que la integridad de la información externa se mantiene. (15)

### 3.4.2. Pruebas de aceptación

El objetivo de estas pruebas es verificar los requisitos, por este motivo, los propios requisitos del sistema son la principal fuente de información a la hora de construir las pruebas de aceptación. Las pruebas de aceptación son creadas a partir de las historias de usuario. Durante una iteración la historia de usuario seleccionada en la planificación de iteraciones se convertirá en una prueba de aceptación. Una historia de usuario puede tener más de una prueba de aceptación, tantas como sean necesarias para garantizar su correcto funcionamiento y no se considera completa hasta que no supera sus pruebas de aceptación. Una prueba de aceptación es como una caja negra. Cada una de ellas representa una salida esperada del sistema. Es responsabilidad del cliente verificar la corrección de las pruebas de aceptación y tomar decisiones a cerca de las mismas. (15) La prueba de aceptación es la que se le aplicó al sistema. Se muestran las mismas en el Anexo 1.

### 3.5. Conclusiones

En el presente capítulo se analizó de forma detallada la estructura de Drupal para un mejor entendimiento del diseño propuesto. Se desarrollaron las tareas correspondientes para dar solución a las historias de usuario y las pruebas de aceptación que propician al cliente conformidad y seguridad ante el sistema. Con el fin de este capítulo se da por terminada la propuesta que trae este trabajo.

### CONCLUSIONES

A través de la implementación del Portal de Calidad de la Facultad 1 para la gestión de la información de los procesos de la calidad en los proyectos productivos de la facultad 1 se le dio solución a los problemas relacionados con la planificación y divulgación de sus actividades, búsqueda de información y mejora del intercambio entre sus miembros, simplificando de esta forma el trabajo del personal. Está provisto de un ambiente cómodo y fácil de entender.

Dada la necesidad del país de emigrar a software libre se utilizó después del estudio de las herramientas y tecnologías como lenguaje de programación PHP, PostgreSQL como gestor de base de datos, el servidor web Apache y como herramienta de desarrollo Zend Studio.

La metodología utilizada fue XP, atendiendo que se estaba en presencia de un proyecto corto y la necesidad del intercambio frecuente con el cliente asegurando de esta forma que las funcionalidades descritas en las historias de usuario satisfagan sus necesidades.



### RECOMENDACIONES

En el presente trabajo se recomienda:

- Trabajar en nuevas mejoras del diseño del portal, teniendo en cuenta la opinión de los usuarios.
- Ampliar las funcionalidades del portal, que posibilite un mejor servicio, para de esta forma satisfacer las necesidades de los usuarios de una manera más óptima.
- Hacer un estudio a fondo de los Sistemas de Gestión de Contenidos en general para de esta manera aprovecharlos en su total plenitud.
- Crear un pequeño grupo para mantener actualizado el portal en colaboración con los aseguradores ya que estos son los que trabajan directamente en el mismo o que sean los mismos aseguradores de cada proyecto incluido el Asesor de Calidad de la facultad.
- Utilizar el sistema de una forma óptima, ya que más que una aplicación es una herramienta que en dependencia de su uso, puede tomar un gran alcance.

### TRABAJOS CITADOS

1. **Cano, Arturo Avila.** Periodismo digital, sociedad de la información, cibercultura, nuevas tecnologías aplicadas a la comunicación. [En línea] 28 de octubre de 2005. [Citado el: 15 de enero de 2009.] <http://seminariojustosierra.blogspot.com/2005/10/definicin-de-portal-pgina-web-sitio.html>.
2. **Alvarez, Rubén.** Lenguajes del lado del servidor o cliente. [desarrolloweb.com](http://www.desarrolloweb.com). [En línea] [Citado el: 15 de enero de 2009.] <http://www.desarrolloweb.com/articulos/239.php>.
3. PostgreSQL. [En línea] [Citado el: 20 de enero de 2009.] <http://www.postgresql.org/>.
4. Hypertext Preprocessor.PHP. . [En línea] [Citado el: 20 de enero de 2009.] <http://www.php.net/>.
5. FAQ sobre APACHE. HTMLPOINT.com. . [En línea] [Citado el: 25 de enero de 2009.] <http://www.htmlpoint.com/faq/apache/01.htm>.
6. Sistema de gestión de contenidos: Artículo: So what is a CMS? . [En línea] [Citado el: 30 de enero de 2009.] [http://typo3.com/What\\_is\\_a\\_CMS.1351.0.html](http://typo3.com/What_is_a_CMS.1351.0.html).
7. Sistemas de Gestión de Contenidos: “About”. Drupal.org. . [En línea] [Citado el: 3 de febrero de 2009.] <http://drupal.org/features>.
8. Sistemas de Gestión de Contenidos: “Module developer's guide”. . [En línea] [Citado el: 5 de febrero de 2009.] <http://drupal.org/node/508>.
9. Sistemas de Gestión de Contenidos: “User's guide”. [En línea] [Citado el: 5 de febrero de 2009.] <http://drupal.org/node/6261>.
10. XP. A gentle introduction. [En línea] [Citado el: 7 de febrero de 2009.] <http://www.extremeprogramming.org/>.
11. **Beck, K.** *Extreme Programming Explained*. s.l. : Addison Wesley, 2000.
12. **Beck, K. y Fowler, M.** *Planeando en Programación Extrema*. 2000.
13. **Booch, G., Rumbaugh, J. y Jacobson.** *El Lenguaje Unificado de Modelado*. 1999.
14. **Crispin, L. y House, T.** *Testing Extreme Programming*. s.l. : Addison Wesley, 2002.
15. **Pressman, R.** *Software Engineering. A Practitioner's Approach*. USA : s.n., 1999.
16. Ventajas e inconvenientes de las aplicaciones web. . [En línea] [Citado el: 11 de febrero de 2009.] <http://www.avidos.net/blogold/aplicaciones-web/>.
17. Hooks. [En línea] [Citado el: 15 de marzo de 2009.] <http://api.drupal.org/api/group/hooks/5..>
18. **LETELIER, P.** *Metodologías ágiles para el desarrollo de software*. 2004.

19. ¿Qué son las bases de datos? . [En línea] 26 de octubre de 2007. [Citado el: 20 de marzo de 2009.] <http://www.maestrosdelweb.com/principiantes/%C2%BFque-son-las-bases-de-datos>.
20. IEEE. (1990). Standard Glossary of Software Engineering Terminology .

### BIBLOGRAFÍA

1. Módulos de Drupal. [En línea] 11 de abril de 2009. <http://drupal.org/handbook/>.
2. ANONIMO. ¿Qué es un Sitio web y Página web? - Definición de Sitio web y Página web. [En línea] [Citado el: 20 de abril de 2009.] <http://www.masadelante.com/faq-sitio-web.htm#contenido>.
3. ARCEO, C. M. Portales: puerta de entrada a la información 2005. [En línea] [http://www.iworld.com.mx/iw\\_news\\_read.asp?iwid=4064](http://www.iworld.com.mx/iw_news_read.asp?iwid=4064).
4. **JOSÉ H. CANÓS, P. L., MARÍA CARMEN PENADÉS.** Metodologías Ágiles en el Desarrollo de Software. [En línea] <http://www.willydev.net/descargas/prev/TodoAgil.Pdf>.
5. Installing contributed modules. [En línea] <http://drupal.org/node/70151>.
6. Aplicaciones Web. [En línea] [Citado el: 20 de abril de 2009.] [http://es.wikipedia.org/wiki/Aplicaci%C3%B3n\\_Web](http://es.wikipedia.org/wiki/Aplicaci%C3%B3n_Web).
7. **Sanchez, María A. Mendoza.** Metodologías de Desarrollo de Software. [En línea] 7 de junio de 2004. [http://www.informatizate.net/articulos/pdfs/metodologias\\_de\\_desarrollo\\_de\\_software\\_07062004.pdf](http://www.informatizate.net/articulos/pdfs/metodologias_de_desarrollo_de_software_07062004.pdf).
8. **MOLPECERES, A.** Procesos de Desarrollo: RUP, XP, FDD. [En línea] <http://tutorial.emagister.com/frame.cfm>.
9. **Cuerda García, Xavier & Minguillón Alfonso, Julia.** Introducción a los Sistemas de Gestión de Contenidos (CMS). [En línea] [Citado el: 3 de mayo de 2009.] <http://mosaic.uoc.edu/articulos/cms1204.html>.
10. CMS ventajas. [En línea] [http://www.cmssoluciones.com/index.php?option=com\\_content&task=view&id=27&Itemid=97](http://www.cmssoluciones.com/index.php?option=com_content&task=view&id=27&Itemid=97).
11. **García Gómez, Juan Carlos & Saorín Pérez, Tomás.** Los Portales de Internet. [En línea] [Citado el: 10 de mayo de 2009.] <http://www.um.es/gtiweb/curso/seis.htm>.
12. Portal (internet) - Wikipedia, la enciclopedia libre. [En línea] junio de 2007. [http://es.wikipedia.org/wiki/Portal\\_\(internet\)](http://es.wikipedia.org/wiki/Portal_(internet)).
13. **SANCHEZ, M. A. M.** *Metodologías De Desarrollo De Software.* 2004.
14. **Valdarrama, Satiago L.** Planificacion Extrema en pocos minutos. . [En línea] <http://svpino.blogspot.com/2005/10/programacin-extrema-en-pocos-minutos.html>.
15. **Reynoso, Carlos.** Metodologías de Desarrollo de Software Ágiles. . [En línea] <http://www.sel.unsl.edu.ar/ApuntesMaes/2004/Metodologias%20Agiles.doc>.

16. **Ferrer Zarzuela, Jorge.** Metodologías Ágiles. . [En línea] <http://libresoft.es/oldsite/downloads/ferrer-20030312.pdf>.
17. **SERRANO, J.** Conceptos Básicos, Drupal . [En línea] [http://www.drupal.org.es/manuales/guia\\_del\\_usuario/conceptos\\_basicos](http://www.drupal.org.es/manuales/guia_del_usuario/conceptos_basicos).
18. **Mellado, Juan.** Instalación. [En línea] 2005. <http://www.inmensia.com/articulos/drupal/instalacion.html> .
19. —. Módulos. [En línea] octubre de 2005. <http://www.inmensia.com/articulos/drupal/pages.html> .
20. —. Temas. [En línea] septiembre de 2005. <http://www.inmensia.com/articulos/drupal/temas.htm>.

# GLOSARIO DE TÉRMINOS

**Apache:** Es el servidor web, disponible sin costo alguno, distribuido con licencia Open Source. Es compatible con sistemas operativos basados en Unix y con Windows.

**API:** Interfaz de Programación de Aplicaciones (Application Programming Interface). Conjunto de funciones y procedimientos (o métodos si se refiere a programación orientada a objetos) que ofrece cierta librería para ser utilizado por otro software como una capa de abstracción.

**Artefacto:** En tecnología, es un dispositivo concebido y fabricado, sea de modo artesanal o industrial, por una o más personas.

**BD:** Conjunto de datos interrelacionados, almacenados con carácter más o menos permanente en la computadora, puede ser considerada una colección de datos variables en el tiempo.

**Bookmarks:** es una agenda de direcciones web. Los programas de navegación por la Red dan la posibilidad de guardar las direcciones de las páginas más interesantes o más visitadas para que, si se quiere volver a visitarlas, baste con hacer clic sobre el bookmark.

**Calidad de Software:** Como primera aproximación es importante diferenciar entre la calidad del PRODUCTO Software y la calidad del PROCESO de desarrollo. No obstante, las metas que se establezcan para la calidad del producto van a determinar las metas a establecer para la calidad del proceso de desarrollo, ya que la calidad del producto va a estar en función de la calidad del proceso de desarrollo. Sin un buen proceso de desarrollo es casi imposible obtener un buen producto.

**Calidad:** Conjunto de propiedades y características de un producto o servicio que le confieren su aptitud para satisfacer unas necesidades explícitas o implícitas. La calidad del software es el grado con el que un sistema, componente o proceso cumple los requerimientos especificados y las necesidades o expectativas del cliente o usuario.

**CASE:** Ingeniería asistida por ordenadores.

**CGI** (Por sus siglas en inglés “**Common Gateway Interface**”) es un método para la transmisión de información hacia un compilador instalado en el servidor. Su función principal es la de añadir una mayor interacción a los documentos web que por medio del HTML se presentan de forma estática.

**CMS: Content Management System** (Sistema de Gestión de Contenidos), son sistemas usados para la construcción de aplicaciones que gestionan contenido.

**CSS:** (Hoja de Estilo en Cascada) Dentro del diseño de páginas de Internet se presenta esta como la vanguardia en cuanto a definición de estilos dentro de las plantillas de diseño.

**GOF:** Gang of Four (Banda de los Cuatro) es el nombre con el que se conoce comúnmente a los autores del libro Patrones de Diseño, el cual hace referencia específicamente al diseño orientado a objetos.

**Grupo de Aseguramiento de la Calidad:** Este grupo se encarga de definir las actividades encaminadas al aseguramiento de la calidad de los proyectos productivos inscritos. Definen un conjunto de actividades o normativas que se deben cumplir en los proyectos que permiten tener un estricto control de la calidad en cada uno de los proyectos desarrollados en la universidad.

**Hook:** Del inglés gancho, palabra usada en la comunidad de Drupal para nombrar los métodos de las interfaces que deben implementar los módulos desarrollados para este CMS.

**HTML:** HyperText Markup Language (lenguaje de marcas hipertextuales), diseñado para estructurar textos y presentarlos en forma de hipertexto, que es el formato estándar de las páginas web.

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

**LDAP:** Lightweight Directory Access Protocol: Protocolo ligero de acceso a directorios.

**Licencia GPL:** La **GNU GPL** (General Public License o licencia pública general) es una licencia creada por la Free Software Foundation orientada principalmente a proteger la libre distribución, modificación y uso de software.

**Microsoft:** Compañía de software más grande del mundo. Fue fundada en 1975 por Paul Allen y Bill Gates. Aunque también se conoce por sus lenguajes de programación y aplicaciones para computadores personales, el éxito sobresaliente de Microsoft se debe a sus sistemas operativos DOS y Windows.

**Módulos:** Un módulo es una colección de funciones que se enlazan dentro de Drupal, ofreciendo funcionalidades adicionales.

**Open Source:** (Código abierto) es el término con el que se conoce al software distribuido y desarrollado libremente.

**PHP Nuke:** PHP-Nuke es un sistema automatizado de noticias basado en la web y sistema de gestión de contenido basado en tecnologías PHP y MySQL.

**PHP:** Professional Home Page Tools es un lenguaje de programación el cual se ejecuta en los servidores web o creación de contenido dinámico para sitios web.

**Plugin:** "Parche" para un programa que le añade características nuevas.

**PostgresSQL:** Gestor de base de datos.

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

**RUP:** Proceso Unificado Racional (Rational Unified Process) es un proceso de desarrollo de software y junto con el Lenguaje Unificado de Modelado UML, constituye la metodología estándar más utilizada para el análisis, implementación y documentación de sistemas orientados a objetos.

**SGBD:** Es el software que permite la utilización y/o la actualización de los datos almacenados en una (o varias) base(s) de datos por uno o varios usuarios desde diferentes puntos de vista y a la vez. **Rol:** Papel, cometido o función que tiene o desempeña que interpreta un actor.

**Software:** Conjunto de programas y procedimientos necesarios para hacer posible la realización de una tarea específica, en contraposición a los componentes físicos del sistema.

**UCI:** Universidad de las Ciencias Informáticas.

**URL:** URL significa Uniform Resource Locator, es decir, localizador uniforme de recurso. Es una secuencia de caracteres, de acuerdo a un formato estándar, que se usa para nombrar recursos, como documentos e imágenes en Internet, por su localización.

**World Wide Web:** Es un sistema de documentos de hipertexto enlazados y accesibles a través de Internet.

**WYSIWYG: What You See Is What You Get** (lo que ves es lo que obtienes). Se aplica a los procesadores de texto y otros editores de texto con formato (como los editores de HTML) que permiten escribir un documento viendo directamente el resultado final, frecuentemente el resultado impreso.



## ANEXOS

### Anexo 1 Casos de prueba

#### Prueba HU11\_P1 Autenticar

##### Caso de Prueba de Aceptación

<b>Código:</b> HU11_P1	<b>Historia de Usuario:</b> Autenticar.
<b>Nombre:</b> Autenticar usuario.	
<b>Descripción:</b> Prueba de funcionalidad de la autenticación del usuario.	
<b>Condiciones de ejecución:</b> El módulo LDAP_Integration debe estar previamente activado y configurado el sub-módulo ldapauth. Los usuarios del sistema deben estar creados y configurados con sus respectivos permisos.	
<b>Entrada / Pasos de Ejecución:</b> Se ingresan nombre de usuario y contraseña y se ejecuta la acción autenticar usuario.	
<b>Resultado esperado:</b> El usuario es autenticado correctamente.	
<b>Evaluación de la prueba:</b> Prueba satisfactoria.	

#### Prueba HU2\_P1 Gestionar documentación

##### Caso de Prueba de Aceptación

<b>Código:</b> HU2_P1	<b>Historia de Usuario:</b> Gestionar documentación.
<b>Nombre:</b> Adicionar documentación.	
<b>Descripción:</b> Prueba de funcionalidad adicionar documentación.	
<b>Condiciones de ejecución:</b> El usuario autenticado debe tener los permisos pertinentes.	
<b>Entrada / Pasos de Ejecución:</b> El usuario procede a la inserción de los datos necesarios, y se ejecuta la acción adicionar documentación.	
<b>Resultado esperado:</b> La documentación es adicionada correctamente.	
<b>Evaluación de la prueba:</b> Prueba satisfactoria.	

#### Prueba HU2\_P2 Gestionar documentación

##### Caso de Prueba de Aceptación

<b>Código:</b> HU2_P2	<b>Historia de Usuario:</b> Gestionar documentación.
<b>Nombre:</b> Modificar documentación.	
<b>Descripción:</b> Prueba de funcionalidad modificar documentación.	
<b>Condiciones de ejecución:</b> El usuario autenticado debe tener los permisos pertinentes.	

<b>Entrada / Pasos de Ejecución:</b> El usuario procede a la modificación de los datos necesarios y se ejecuta la acción modificar documentación.
<b>Resultado esperado:</b> La documentación es modificada sin errores.
<b>Evaluación de la prueba:</b> Prueba satisfactoria.

#### Prueba HU2\_P3 Gestionar documentación

##### Caso de Prueba de Aceptación

<b>Código:</b> HU2_P3	<b>Historia de Usuario:</b> Gestionar documentación.
<b>Nombre:</b> Eliminar documentación.	
<b>Descripción:</b> Prueba de funcionalidad eliminar documentación.	
<b>Condiciones de ejecución:</b> El usuario autenticado debe tener los permisos pertinentes.	
<b>Entrada / Pasos de Ejecución:</b> El usuario procede a la eliminación de la información de proyecto y se ejecuta la acción eliminar documentación.	
<b>Resultado esperado:</b> La documentación es eliminada correctamente.	
<b>Evaluación de la prueba:</b> Prueba satisfactoria.	

#### Prueba HU2\_P4 Gestionar documentación

##### Caso de Prueba de Aceptación

<b>Código:</b> HU2_P4	<b>Historia de Usuario:</b> Gestionar documentación.
<b>Nombre:</b> Mostrar documentación.	
<b>Descripción:</b> Prueba de funcionalidad de mostrar documentación.	
<b>Condiciones de ejecución:</b> Debe estar creada previamente la página que permite el proceso.	
<b>Entrada / Pasos de Ejecución:</b> El usuario accede a la sección documentación. Se ejecuta la acción mostrar documentación.	
<b>Resultado esperado:</b> La documentación se muestra correctamente.	
<b>Evaluación de la prueba:</b> Prueba satisfactoria.	

#### Prueba HU3\_P1 Gestionar información de proyecto

##### Caso de Prueba de Aceptación

<b>Código:</b> HU3_P1	<b>Historia de Usuario:</b> Gestionar información de proyecto.
<b>Nombre:</b> Adicionar información de proyecto.	
<b>Descripción:</b> Prueba de funcionalidad adicionar información de proyecto.	
<b>Condiciones de ejecución:</b> El usuario autenticado debe tener los permisos pertinentes.	
<b>Entrada / Pasos de Ejecución:</b> El usuario procede a la inserción de los datos necesarios, y se ejecuta la	

acción adicionar información de proyecto.
<b>Resultado esperado:</b> La información de proyecto es adicionada correctamente.
<b>Evaluación de la prueba:</b> Prueba satisfactoria.

### Prueba HU3\_P2 Gestionar información de proyecto

#### Caso de Prueba de Aceptación

<b>Código:</b> HU3_P2	<b>Historia de Usuario:</b> Gestionar información de proyecto.
<b>Nombre:</b> Modificar información de proyecto.	
<b>Descripción:</b> Prueba de funcionalidad modificar información de proyecto.	
<b>Condiciones de ejecución:</b> El usuario autenticado debe tener los permisos pertinentes.	
<b>Entrada / Pasos de Ejecución:</b> El usuario procede a la modificación de los datos necesarios y se ejecuta la acción modificar información de proyecto.	
<b>Resultado esperado:</b> La información de proyecto es modificada sin errores.	
<b>Evaluación de la prueba:</b> Prueba satisfactoria.	

### Prueba HU3\_P3 Gestionar información de proyecto

#### Caso de Prueba de Aceptación

<b>Código:</b> HU3_P3	<b>Historia de Usuario:</b> Gestionar información de proyecto.
<b>Nombre:</b> Eliminar información de proyecto.	
<b>Descripción:</b> Prueba de funcionalidad eliminar información de proyecto.	
<b>Condiciones de ejecución:</b> El usuario autenticado debe tener los permisos pertinentes.	
<b>Entrada / Pasos de Ejecución:</b> El usuario procede a la eliminación de la información de proyecto y se ejecuta la acción eliminar información de proyecto.	
<b>Resultado esperado:</b> La información de proyecto es eliminada correctamente.	
<b>Evaluación de la prueba:</b> Prueba satisfactoria.	

### Prueba HU3\_P4 Gestionar información de proyecto

#### Caso de Prueba de Aceptación

<b>Código:</b> HU3_P4	<b>Historia de Usuario:</b> Gestionar información de proyecto.
<b>Nombre:</b> Mostrar información de proyecto.	
<b>Descripción:</b> Prueba de funcionalidad de mostrar información de proyecto.	
<b>Condiciones de ejecución:</b> Debe estar creada previamente la vista que permite el proceso.	
<b>Entrada / Pasos de Ejecución:</b> El usuario accede a la sección ver información de proyecto. Se ejecuta la acción mostrar información de proyecto.	

<b>Resultado esperado:</b> La información de proyecto se muestra correctamente.
<b>Evaluación de la prueba:</b> Prueba satisfactoria.

#### Prueba HU4\_P1 Gestionar miembro por proyecto

##### Caso de Prueba de Aceptación

<b>Código:</b> HU4_P1	<b>Historia de Usuario:</b> Gestionar miembro por proyecto.
<b>Nombre:</b> Adicionar listado de miembro por proyecto.	
<b>Descripción:</b> Prueba de funcionalidad adicionar listado de miembro por proyecto.	
<b>Condiciones de ejecución:</b> El usuario autenticado debe tener los permisos pertinentes.	
<b>Entrada / Pasos de Ejecución:</b> El usuario procede a la inserción de los datos necesarios, y se ejecuta la acción adicionar listado de miembro por proyecto.	
<b>Resultado esperado:</b> El listado de miembro por proyecto es adicionado correctamente.	
<b>Evaluación de la prueba:</b> Prueba satisfactoria.	

#### Prueba HU4\_P2 Gestionar miembro por proyecto

##### Caso de Prueba de Aceptación

<b>Código:</b> HU4_P2	<b>Historia de Usuario:</b> Gestionar miembro por proyecto.
<b>Nombre:</b> Modificar listado de miembro por proyecto.	
<b>Descripción:</b> Prueba de funcionalidad modificar listado de miembro por proyecto.	
<b>Condiciones de ejecución:</b> El usuario autenticado debe tener los permisos pertinentes.	
<b>Entrada / Pasos de Ejecución:</b> El usuario procede a la modificación de los datos necesarios y se ejecuta la acción modificar listado de miembro por proyecto.	
<b>Resultado esperado:</b> El listado de miembro por proyecto es modificado sin errores.	
<b>Evaluación de la prueba:</b> Prueba satisfactoria.	

#### Prueba HU4\_P3 Gestionar miembro por proyecto

##### Caso de Prueba de Aceptación

<b>Código:</b> HU4_P3	<b>Historia de Usuario:</b> Gestionar miembro por proyecto.
<b>Nombre:</b> Eliminar listado de miembro por proyecto.	
<b>Descripción:</b> Prueba de funcionalidad eliminar listado de miembro por proyecto.	
<b>Condiciones de ejecución:</b> El usuario autenticado debe tener los permisos pertinentes.	
<b>Entrada / Pasos de Ejecución:</b> El usuario procede a la eliminación del listado de miembro por proyecto y se ejecuta la acción eliminar listado de miembro por proyecto.	
<b>Resultado esperado:</b> El listado de miembro por proyecto es eliminado correctamente.	

<b>Evaluación de la prueba:</b> Prueba satisfactoria.
---

#### Prueba HU4\_P4 Gestionar miembro por proyecto

##### Caso de Prueba de Aceptación

<b>Código:</b> HU4_P4	<b>Historia de Usuario:</b> Gestionar miembro por proyecto.
<b>Nombre:</b> Mostrar listado de miembro por proyecto.	
<b>Descripción:</b> Prueba de funcionalidad de mostrar listado de miembro por proyecto.	
<b>Condiciones de ejecución:</b> Debe estar creada previamente la página que permite el proceso.	
<b>Entrada / Pasos de Ejecución:</b> El usuario accede a la sección de grupo de aseguradores. Se ejecuta la acción mostrar listado de miembro por proyecto.	
<b>Resultado esperado:</b> El listado de miembro por proyecto se muestra correctamente.	
<b>Evaluación de la prueba:</b> Prueba satisfactoria.	

#### Prueba HU5\_P1 Gestionar noticias

##### Caso de Prueba de Aceptación

<b>Código:</b> HU5_P1	<b>Historia de Usuario:</b> Gestionar noticias.
<b>Nombre:</b> Adicionar noticia.	
<b>Descripción:</b> Prueba de funcionalidad agregar noticia.	
<b>Condiciones de ejecución:</b> El usuario autenticado debe tener los permisos pertinentes.	
<b>Entrada / Pasos de Ejecución:</b> El usuario procede a la inserción de los datos necesarios, especificando el tipo de noticia a agregar y se ejecuta la acción agregar noticia.	
<b>Resultado esperado:</b> La noticia es agregada correctamente.	
<b>Evaluación de la prueba:</b> Prueba satisfactoria.	

#### Prueba HU5\_P2 Gestionar noticias

##### Caso de Prueba de Aceptación

<b>Código:</b> HU5_P2	<b>Historia de Usuario:</b> Gestionar noticias.
<b>Nombre:</b> Modificar noticia.	
<b>Descripción:</b> Prueba de funcionalidad modificar noticia.	
<b>Condiciones de ejecución:</b> El usuario autenticado debe tener los permisos pertinentes.	
<b>Entrada / Pasos de Ejecución:</b> El usuario procede a la modificación de los datos necesarios y se ejecuta la acción modificar noticia.	
<b>Resultado esperado:</b> La noticia es modificada correctamente.	
<b>Evaluación de la prueba:</b> Prueba satisfactoria.	

**Prueba HU5\_P3 Gestionar noticias****Caso de Prueba de Aceptación**

<b>Código:</b> HU5_P3	<b>Historia de Usuario:</b> Gestionar noticias.
<b>Nombre:</b> Eliminar noticia.	
<b>Descripción:</b> Prueba de funcionalidad de eliminar noticia.	
<b>Condiciones de ejecución:</b> El usuario autenticado debe tener los permisos pertinentes.	
<b>Entrada / Pasos de Ejecución:</b> El usuario procede a la eliminación de la noticia y se ejecuta la acción eliminar noticia.	
<b>Resultado esperado:</b> La noticia es eliminada correctamente.	
<b>Evaluación de la prueba:</b> Prueba satisfactoria.	

**Prueba HU5\_P4 Gestionar noticias****Caso de Prueba de Aceptación**

<b>Código:</b> HU5_P4	<b>Historia de Usuario:</b> Gestionar noticias.
<b>Nombre:</b> Mostrar noticias.	
<b>Descripción:</b> Prueba de funcionalidad mostrar noticias.	
<b>Condiciones de ejecución:</b> Deben estar creadas previamente las vistas que permiten el proceso.	
<b>Entrada / Pasos de Ejecución:</b> El usuario accede a la sección noticias y selecciona el tipo de noticia a visualizar. Se ejecuta la acción mostrar noticias.	
<b>Resultado esperado:</b> Las noticias se muestran satisfactoriamente.	
<b>Evaluación de la prueba:</b> Prueba satisfactoria.	

**Prueba HU5\_P5 Gestionar noticias****Caso de Prueba de Aceptación**

<b>Código:</b> HU5_P5	<b>Historia de Usuario:</b> Gestionar noticias.
<b>Nombre:</b> Mostrar noticia.	
<b>Descripción:</b> Prueba de funcionalidad mostrar noticia.	
<b>Condiciones de ejecución:</b> Deben estar mostradas las noticias previamente.	
<b>Entrada / Pasos de Ejecución:</b> El usuario procede a la selección de la noticia que desee visualizar y se ejecuta la acción mostrar noticia.	
<b>Resultado esperado:</b> La noticia seleccionada se muestra correctamente.	
<b>Evaluación de la prueba:</b> Prueba satisfactoria.	

**Prueba HU8\_P1 Gestionar enlaces****Caso de Prueba de Aceptación**

<b>Código:</b> HU8_P1	<b>Historia de Usuario:</b> Gestionar enlaces.
<b>Nombre:</b> Adicionar enlace.	
<b>Descripción:</b> Prueba de funcionalidad agregar enlace	
<b>Condiciones de ejecución:</b> El usuario autenticado debe tener los permisos pertinentes.	
<b>Entrada / Pasos de Ejecución:</b> El usuario procede a insertar los datos necesarios y se ejecuta la acción agregar enlace.	
<b>Resultado esperado:</b> El enlace es agregado correctamente.	
<b>Evaluación de la prueba:</b> Prueba satisfactoria.	

**Prueba HU8\_P2 Gestionar enlaces****Caso de Prueba de Aceptación**

<b>Código:</b> HU8_P2	<b>Historia de Usuario:</b> Gestionar enlaces.
<b>Nombre:</b> Modificar enlace.	
<b>Descripción:</b> Prueba de funcionalidad modificar enlace	
<b>Condiciones de ejecución:</b> El usuario autenticado debe tener los permisos pertinentes.	
<b>Entrada / Pasos de Ejecución:</b> El usuario procede a modificar los datos necesarios y se ejecuta la acción modificar enlace.	
<b>Resultado esperado:</b> El enlace es modificado correctamente.	
<b>Evaluación de la prueba:</b> Prueba satisfactoria.	

**Prueba HU8\_P3 Gestionar enlaces****Caso de Prueba de Aceptación**

<b>Código:</b> HU8_P3	<b>Historia de Usuario:</b> Gestionar enlaces.
<b>Nombre:</b> Eliminar enlace.	
<b>Descripción:</b> Prueba de funcionalidad eliminar enlace	
<b>Condiciones de ejecución:</b> El usuario autenticado debe tener los permisos pertinentes.	
<b>Entrada / Pasos de Ejecución:</b> El usuario procede a eliminar el enlace y se ejecuta la acción eliminar enlace.	
<b>Resultado esperado:</b> El enlace es eliminado correctamente.	
<b>Evaluación de la prueba:</b> Prueba satisfactoria.	

**Prueba HU8\_P4 Gestionar enlaces****Caso de Prueba de Aceptación**

<b>Caso de Prueba de Aceptación</b>	
-------------------------------------	--

<b>Código:</b> HU8_P4	<b>Historia de Usuario:</b> Gestionar enlaces.
<b>Nombre:</b> Mostrar enlaces.	
<b>Descripción:</b> Prueba de funcionalidad mostrar enlaces	
<b>Condiciones de ejecución:</b> Los enlaces deben estar activados.	
<b>Entrada / Pasos de Ejecución:</b> El usuario accede a la sección enlaces y se visualizan los enlaces a otros sitios.	
<b>Resultado esperado:</b> Los enlaces son mostrados correctamente.	
<b>Evaluación de la prueba:</b> Prueba satisfactoria.	

#### Prueba HU8\_P5 Gestionar enlaces

##### Caso de Prueba de Aceptación

<b>Código:</b> HU8_P5	<b>Historia de Usuario:</b> Gestionar enlaces.
<b>Nombre:</b> Mostrar contenido del enlace.	
<b>Descripción:</b> Prueba de funcionalidad mostrar contenido del enlace.	
<b>Condiciones de ejecución:</b> Se debe acceder a un enlace previamente.	
<b>Entrada / Pasos de Ejecución:</b> El usuario accede a un enlace deseado y se visualiza el contenido del enlace.	
<b>Resultado esperado:</b> Los contenidos de los enlaces son mostrados correctamente.	
<b>Evaluación de la prueba:</b> Prueba satisfactoria.	

#### Prueba HU9\_P1 Buscar información

##### Caso de Prueba de Aceptación

<b>Código:</b> HU9_P1	<b>Historia de Usuario:</b> Buscar información.
<b>Nombre:</b> Buscar contenido.	
<b>Descripción:</b> Prueba de funcionalidad buscar contenido	
<b>Condiciones de ejecución:</b> El usuario debe tener los permisos de acceso a este servicio.	
<b>Entrada / Pasos de Ejecución:</b> El usuario procede a ingresar la palabra clave relacionada con el contenido a buscar y se ejecuta la acción buscar contenido.	
<b>Resultado esperado:</b> La búsqueda es realizada correctamente.	
<b>Evaluación de la prueba:</b> Prueba satisfactoria.	

#### Prueba HU9\_P2 Buscar información

##### Caso de Prueba de Aceptación

<b>Código:</b> HU9_P2	<b>Historia de Usuario:</b> Buscar información.
<b>Nombre:</b> Mostrar contenidos encontrados.	



<b>Descripción:</b> Prueba de funcionalidad mostrar contenidos encontrados.
<b>Condiciones de ejecución:</b> El usuario debe ingresar la palabra clave previamente
<b>Entrada / Pasos de Ejecución:</b> El usuario visualiza el resultado de la búsqueda realizada.
<b>Resultado esperado:</b> Los contenidos encontrados se muestran correctamente.
<b>Evaluación de la prueba:</b> Prueba satisfactoria.

**Prueba HU10\_P1 Imprimir información  
Caso de Prueba de Aceptación**

<b>Código:</b> HU10_P1	<b>Historia de Usuario:</b> Imprimir información.
<b>Nombre:</b> Imprimir contenido.	
<b>Descripción:</b> Prueba de funcionalidad Imprimir información	
<b>Condiciones de ejecución:</b> El usuario debe tener los permisos de acceso a este servicio.	
<b>Entrada / Pasos de Ejecución:</b> El usuario procede a seleccionar el contenido a imprimir y se ejecuta la acción Imprimir contenido.	
<b>Resultado esperado:</b> La impresión de documento es realizada correctamente.	
<b>Evaluación de la prueba:</b> Prueba satisfactoria.	

**Prueba HU12\_P1 Gestionar cronograma de revisiones  
Caso de Prueba de Aceptación**

<b>Código:</b> HU12_P1	<b>Historia de Usuario:</b> Gestionar cronograma de revisiones.
<b>Nombre:</b> Adicionar cronograma de revisiones.	
<b>Descripción:</b> Prueba de funcionalidad adicionar cronograma de revisiones.	
<b>Condiciones de ejecución:</b> El usuario autenticado debe tener los permisos pertinentes.	
<b>Entrada / Pasos de Ejecución:</b> El usuario procede a la inserción de los datos necesarios, y se ejecuta la acción adicionar cronograma de revisiones.	
<b>Resultado esperado:</b> El cronograma de revisiones es adicionada correctamente.	
<b>Evaluación de la prueba:</b> Prueba satisfactoria.	

**Prueba HU12\_P2 Gestionar cronograma de revisiones  
Caso de Prueba de Aceptación**

<b>Código:</b> HU12_P2	<b>Historia de Usuario:</b> Gestionar cronograma de revisiones.
<b>Nombre:</b> Modificar cronograma de revisiones.	
<b>Descripción:</b> Prueba de funcionalidad modificar cronograma de revisiones.	
<b>Condiciones de ejecución:</b> El usuario autenticado debe tener los permisos pertinentes.	

<b>Entrada / Pasos de Ejecución:</b> El usuario procede a la modificación de los datos necesarios y se ejecuta la acción modificar cronograma de revisiones.
<b>Resultado esperado:</b> El cronograma de revisiones es modificado sin errores.
<b>Evaluación de la prueba:</b> Prueba satisfactoria.

#### Prueba HU12\_P3 Gestionar cronograma de revisiones

##### Caso de Prueba de Aceptación

<b>Código:</b> HU12_P3	<b>Historia de Usuario:</b> Gestionar cronograma de revisiones.
<b>Nombre:</b> Eliminar cronograma de revisiones.	
<b>Descripción:</b> Prueba de funcionalidad eliminar cronograma de revisiones.	
<b>Condiciones de ejecución:</b> El usuario autenticado debe tener los permisos pertinentes.	
<b>Entrada / Pasos de Ejecución:</b> El usuario procede a la eliminación del cronograma de revisiones y se ejecuta la acción eliminar cronograma de revisiones.	
<b>Resultado esperado:</b> El cronograma de revisiones se elimina correctamente.	
<b>Evaluación de la prueba:</b> Prueba satisfactoria.	

#### Prueba HU12\_P4 Gestionar cronograma de revisiones

##### Caso de Prueba de Aceptación

<b>Código:</b> HU12_P4	<b>Historia de Usuario:</b> Gestionar cronograma de revisiones.
<b>Nombre:</b> Mostrar cronograma de revisiones.	
<b>Descripción:</b> Prueba de funcionalidad de mostrar cronograma de revisiones.	
<b>Condiciones de ejecución:</b> Debe estar creada previamente la página que permite el proceso.	
<b>Entrada / Pasos de Ejecución:</b> El usuario accede a la sección cronograma de revisiones. Se ejecuta la acción mostrar cronograma de revisiones.	
<b>Resultado esperado:</b> El cronograma de revisiones se muestra correctamente.	
<b>Evaluación de la prueba:</b> Prueba satisfactoria.	

#### Prueba HU13\_P1 Gestionar preguntas frecuentes

##### Caso de Prueba de Aceptación

<b>Código:</b> HU13_P1	<b>Historia de Usuario:</b> Gestionar preguntas frecuentes.
<b>Nombre:</b> Adicionar preguntas frecuentes.	
<b>Descripción:</b> Prueba de funcionalidad adicionar preguntas frecuentes.	
<b>Condiciones de ejecución:</b> El usuario autenticado debe tener los permisos pertinentes.	
<b>Entrada / Pasos de Ejecución:</b> El usuario procede a la inserción de los datos necesarios, y se ejecuta la	

acción adicionar preguntas frecuentes.
<b>Resultado esperado:</b> Las preguntas frecuentes son adicionas correctamente.
<b>Evaluación de la prueba:</b> Prueba satisfactoria.

#### Prueba HU13\_P2 Gestionar preguntas frecuentes

##### Caso de Prueba de Aceptación

<b>Código:</b> HU13_P2	<b>Historia de Usuario:</b> Gestionar preguntas frecuentes.
<b>Nombre:</b> Modificar preguntas frecuentes.	
<b>Descripción:</b> Prueba de funcionalidad modificar preguntas frecuentes.	
<b>Condiciones de ejecución:</b> El usuario autenticado debe tener los permisos pertinentes.	
<b>Entrada / Pasos de Ejecución:</b> El usuario procede a la modificación de los datos necesarios y se ejecuta la acción modificar preguntas frecuentes.	
<b>Resultado esperado:</b> Las preguntas frecuentes se modifican sin errores.	
<b>Evaluación de la prueba:</b> Prueba satisfactoria.	

#### Prueba HU13\_P3 Gestionar preguntas frecuentes

##### Caso de Prueba de Aceptación

<b>Código:</b> HU13_P3	<b>Historia de Usuario:</b> Gestionar preguntas frecuentes.
<b>Nombre:</b> Eliminar preguntas frecuentes.	
<b>Descripción:</b> Prueba de funcionalidad eliminar preguntas frecuentes.	
<b>Condiciones de ejecución:</b> El usuario autenticado debe tener los permisos pertinentes.	
<b>Entrada / Pasos de Ejecución:</b> El usuario procede a la eliminación del preguntas frecuentes y se ejecuta la acción eliminar preguntas frecuentes.	
<b>Resultado esperado:</b> Las preguntas frecuentes se eliminan correctamente.	
<b>Evaluación de la prueba:</b> Prueba satisfactoria.	

#### Prueba HU13\_P4 Gestionar preguntas frecuentes

##### Caso de Prueba de Aceptación

<b>Código:</b> HU13_P4	<b>Historia de Usuario:</b> Gestionar preguntas frecuentes.
<b>Nombre:</b> Mostrar preguntas frecuentes.	
<b>Descripción:</b> Prueba de funcionalidad de mostrar preguntas frecuentes.	
<b>Condiciones de ejecución:</b> Debe estar creada previamente la página que permite el proceso.	
<b>Entrada / Pasos de Ejecución:</b> El usuario accede a la sección preguntas frecuentes. Se ejecuta la acción mostrar preguntas frecuentes.	

<b>Resultado esperado:</b> Las preguntas frecuentes se muestran correctamente.
<b>Evaluación de la prueba:</b> Prueba satisfactoria.

**Prueba HU14\_P1 Mostrar el mapa de navegación**

**Caso de Prueba de Aceptación**

<b>Código:</b> HU14_P1	<b>Historia de Usuario:</b> Mostrar el mapa de navegación.
<b>Nombre:</b> Mostrar mapa web.	
<b>Descripción:</b> Prueba de funcionalidad mostrar mapa web.	
<b>Condiciones de ejecución:</b> Debe estar activado el módulo Site map.	
<b>Entrada / Pasos de Ejecución:</b> El usuario accede al ícono del mapa y se ejecuta la acción mostrar mapa web.	
<b>Resultado esperado:</b> El mapa web es mostrado correctamente.	
<b>Evaluación de la prueba:</b> Prueba satisfactoria.	