

**UNIVERSIDAD DE LAS CIENCIAS INFORMÁTICAS**

**FACULTAD 8**



**“Sistema de Control de Estado de las Especificaciones de Casos de Uso y  
Manuales de Usuario del SIIPOL”**



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

**Autores:**

Ivonne Isabel Guerra Anaya

Alexey Casanova Alvarez

**Tutores:**

Ing. Irina Cancela Nieto

Ing. Liana Toledo Bueno

Ciudad de La Habana, junio 2010

# DECLARACIÓN DE AUTORÍA

Declaramos que somos los únicos autores del trabajo “**Sistema de Control de Estado de las Especificaciones de Casos de Uso y Manuales de Usuario del SIIPOL**” y autorizamos a la Facultad 8 de la Universidad de las Ciencias Informáticas a hacer uso del mismo en su beneficio.

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

Autores:

\_\_\_\_\_

Alexey Casanova Alvarez

\_\_\_\_\_

Ivonne Isabel Guerra Anaya

Tutores:

\_\_\_\_\_

Irina Cancela Nieto

\_\_\_\_\_

Liana Toledo Bueno



*“El futuro de nuestra patria tiene que ser necesariamente un futuro de hombres de ciencia, tiene que ser un futuro de hombres de pensamiento, porque precisamente es lo que más estamos sembrando; lo que más estamos sembrando son oportunidades a la inteligencia.”*

## DEDICATORIA

*A mi mamá por ser mi mayor inspiración y ejemplo a seguir. A la que le debo todo lo que he logrado por el empeño que ha puesto en hacer de mí una mejor persona.*

*A mi papá por creer siempre en mí y darme todo su amor en cada momento. Por ser el hombre más importante de mi vida.*

*A mis hermanitos Caty y Ernesto, a los que quiero mucho y espero poder servirle de ejemplo.*

*A mi abuela Tata, que la quiero con todo mi corazón y aún no es suficiente. Por haberme educado y brindado su cariño durante toda mi vida.*

*A toda mi familia que siempre ha estado pendiente de cada uno de mis pasos.*

### **Ivonne**

*A mi mamá porque sin su confianza y apoyo no hubiera podido llegado hasta este momento de mi vida.*

*A mi papá por brindarme sus consejos, que fueron muy útiles en toda mi vida.*

*A mis hermanos Alex y Aliannis por ser el mejor ejemplo a seguir, por ser ellos lo mejor que Dios me pudo dar.*

*A mis sobrinos de manera general, en especial a mi sobrina Dariannis.*

*A mis abuelitos Zeida y Paquito que los quiero con toda mi vida.*

*A mi novia por todo el amor brindado en estos 5 años de carrera universitaria.*

*A toda mi familia de manera general.*

**Alexey**

## AGRADECIMIENTOS

*A la Revolución Cubana por darnos la oportunidad de hacer realidad el sueño de  
convertirnos en profesionales.*

*A la Universidad de las Ciencias Informáticas por formarnos como buenos ingenieros*

*A toda nuestra familia que es nuestro mayor sostén en cada uno de nuestros pasos.*

*A los profesores que nos ayudaron a perfeccionar nuestro trabajo.*

*A los amigos que siempre nos apoyaron en los momentos difíciles.*

*A todos los que nos han ayudado a convertir nuestros sueños en realidad,*

*“Muchas Gracias”*

*Ivonne y Alexey*

*A mi mamá que siempre se ha esforzado por hacer cumplir cada uno de mis sueños.*

*A mi papá que siempre ha tenido confianza en mí y me ha brindado todo su apoyo*

*A mi abuela Tata que me quiere con la vida y siempre espera lo mejor de mí.*

*A mis hermanitos, que son mi tesoro.*

*A Carrasco que ha formado parte de mi familia y siempre me ha ayudado, apoyándome de forma incondicional.*

*A Iraida que siempre se ha preocupado por mí y me ha ayudado en todo lo que he necesitado.*

*A mi bisabuela Viva, que me educó desde pequeña con todo el amor del mundo.*

*A Pedri, que siempre ha estado dispuesto a darme lo mejor y a hacer cualquier cosa por mí.*

*A mi compañero de tesis y aventuras Alexey, que en esta difícil tarea me ha dado la fuerza suficiente para seguir adelante sin importar los obstáculos.*

*A mis amigas y amigos de Banes y de la UCI, que siempre me han ayudado en todo lo que he necesitado, especialmente mis amigas, a las que quiero muchísimo y me han dado valiosos consejos. Gracias a todos ellos he aprendido que nada es imposible.*

*A mis compañeros de aula, que me han acompañado en este largo camino.*

*A todos los que me han brindado su apoyo en cada momento de mi vida.*

***Ivonne***

*Le doy gracias a Dios por permitirme haber logrado mi sueño.*

*En especial a mi mami por su gran amor incondicional, por apoyarme en todos los momentos de mi vida y por depositar toda su confianza en mí.*

*A mis hermanos Alexander, Aliannis y Adisnelvis y a mi padre por haber sido el motor impulsor de mi vida, sobre todo en mi carrera universitaria y por haberme permitido la posibilidad de hacer realidad mi sueño.*

*A toda mi familia de manera general, por haber estado pendientes de mí.*

*A mi novia por estar siempre presente cuando más lo necesitaba y que siempre me apoyó en todo momento.*

*Le agradezco mucho a mi compañera de tesis, por su paciencia conmigo, por ayudarme y por haberme acompañado a lograr este objetivo en mi vida.*

*A mis compañeros de estudio que estuvieron presente siempre que los necesitaba apoyándome, permitiendo que mi vida universitaria fuera más fácil.*

*A todos aquellos que de una forma u otra me han ayudado a lograr esta meta.*

*Alexey*

## **Resumen**

Durante el proceso de desarrollo de un software es importante tener dominio del grado de avance del mismo. Para esto, es necesario controlar eficientemente, todas las actividades que se desarrollan en cada etapa. En este trabajo se propuso la creación del “**Sistema de Control de Estado de las Especificaciones de Casos de Uso y Manuales de Usuario del SIIPOL**”, a través del cual se esperó obtener una mejora considerable en los mecanismos de comunicación dentro del proyecto CICPC, en relación con las actividades que se podían realizar a partir de cada Caso de Uso y Manual de Usuario como entidades fundamentales.

La investigación que se realizó recogió cada una de las etapas del proceso de desarrollo de la solución propuesta, tomando en cuenta los principios sugeridos por la metodología XP, como guía del desarrollo. El proceso de construcción de la herramienta se culminó con la firma del Acta de Aceptación del producto en conjunto con el cliente, lo que validó la conformidad y efectividad de la herramienta construida. Con lo cual se le dio cumplimiento al objetivo general de la investigación aportando a través del nuevo sistema implementado, mayor eficiencia en el control del trabajo realizado por los jefes de equipo del proyecto, y de esta forma se posibilitó mejorar los procesos organizativos de la información referente a los Casos de Uso y Manuales de Usuario del proyecto CICPC.

**Palabras Clave:** control, estado, Especificación de Caso de Uso, Manual de Usuario.



## Índice de Contenido.

Introducción.....	1
CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA.....	5
Introducción.....	5
1.1. Sistemas existentes vinculados al campo de acción:.....	5
1.1.1. TRAC.....	5
1.1.2. REDMINE.....	6
1.1.3. E-GROUPWARE.....	7
1.2. Tendencias y tecnologías actuales.....	8
1.2.1. Aplicaciones Web.....	9
1.2.2. Servidores Web.....	9
1.2.2.1. Internet Information Services (IIS).....	9
1.2.2.2. Apache.....	10
1.2.3. Lenguajes de Programación.....	10
1.2.3.1. Java.....	11
1.2.3.2. Hypertext Preprocessor (PHP).....	11
1.2.3.3. Fundamentación del lenguaje de programación a utilizar.....	12
1.2.4. Sistema Gestores de Base de Datos.....	12
1.2.4.1. MySQL.....	13
1.2.4.2. PostgreSQL.....	13
1.2.4.3. Fundamentación del Sistema de Base de Datos a utilizar.....	14
1.2.5. Metodología de Desarrollo de Software.....	15
1.2.5.1. Rational Unified Process (RUP).....	15
1.2.5.2. Extreme Programming (XP).....	17
1.2.5.3. Fundamentación de la metodología de desarrollo a utilizar.....	20
1.2.6. Sistema de Gestión de Contenidos.....	20
1.2.6.1. CMS Drupal.....	21
1.2.6.2. CMS Joomla.....	21
1.2.6.3. Fundamentación de la selección del CMS a utilizar.....	22
1.2.7. Otros lenguajes utilizados.....	22
1.2.7.1. Hojas de estilo en cascada (CCS).....	22

1.2.7.2. Java Script.....	23
1.2.8. Otras herramientas de apoyo para el desarrollo. ....	23
1.2.8.1. Zend Studio. ....	23
1.2.8.2. Aptana. ....	23
1.2.8.3. Dreamweaver. ....	23
1.2.8.4. NuSphere. ....	24
Conclusiones.....	24
<b>CAPÍTULO 2: PROPUESTA DE SOLUCIÓN.....</b>	<b>25</b>
Introducción.....	25
2.1. Descripción del Flujo Actual de los Procesos.....	25
2.2. Propuesta de Sistema.....	26
2.3. Usuarios del Sistema. ....	27
2.4. Requisitos del Software. ....	28
2.4.1. Funcionalidades del sistema.....	28
2.4.2. Otras características del sistema. ....	34
Conclusiones.....	35
<b>CAPÍTULO 3: EXPLORACIÓN Y PLANIFICACIÓN.....</b>	<b>36</b>
Introducción.....	36
3.1. Fase de Exploración. ....	36
3.1.1. Historias de Usuario. ....	36
3.2. Fase de Planificación.....	47
3.2.1 Estimación del esfuerzo por Historias de Usuario.....	48
3.2.2. Plan de Iteraciones. ....	49
3.2.3. Plan de Duración de las Iteraciones.....	50
3.2.4. Plan de Entrega.....	51
Conclusiones.....	51
<b>CAPÍTULO 4: DISEÑO E IMPLEMENTACIÓN.....</b>	<b>52</b>
Introducción.....	52

4.1.1	Diseño de la Base de Datos.....	56
4.2.	Fase de Implementación.....	57
	Conclusiones.....	58
	CAPÍTULO 5: VALIDACIÓN DE LA PROPUESTA.....	59
	Introducción.....	59
4.4.1.	Pruebas Realizadas.....	59
4.4.2.	Resultados de las Pruebas.....	59
4.4.3.	Valoración del producto.....	61
	Conclusiones.....	63
	CONCLUSIONES GENERALES.....	64
	RECOMENDACIONES.....	65
	REFERENCIAS BIBLIOGRÁFICAS:.....	66
	BIBLIOGRAFÍA:.....	68
	GLOSARIO DE TÉRMINOS.....	69
	ANEXOS.....	70
	Anexo 1: Diagrama del proceso a automatizar.....	70
	Anexo 2: Tareas de Implementación de la primera iteración.....	71
	Anexo 3: Tareas de Implementación de la segunda iteración.....	74
	Anexo 4. Tareas de Implementación de la tercera iteración.....	76
	Anexo 5. Casos de Prueba de Aceptación.....	78
	Anexo 6. Acta de Aceptación del producto de software realizado.....	87

## Índice de Figuras.

Figura 1. Flujos de Trabajo de RUP.....	16
Figura 2: Fases de un proyecto en XP.....	18
Figura 3. Etapas de XP.....	19
Figura 4. Propuesta del Sistema.....	26
Figura 5. Capas de Drupal.....	27
Figura 6. Diagrama de clases de diseño de Drupal.....	52
Figura 7. Nuevos módulos implementados.....	53
Figura 8. Diagrama Entidad-Relación de la BD.....	57
Figura 9. Diagrama de Despliegue.....	58
Figura 10. Cantidad de no conformidades por iteración.....	60
Figura 11. Clasificación de las NC por tipo.....	61
Figura 12. Puntuación por pregunta dada por los encuestados.....	62
Figura 13. Total de puntos por cada criterio evaluado en cada pregunta.....	63

## Índice de Tablas.

Tabla 1. Funciones de los usuarios del sistema.....	28
Tabla 2. HU Gestionar Usuario.....	36
Tabla 3. HU Gestionar Rol de Usuario.....	37
Tabla 4. HU Autenticar Usuario.....	37
Tabla 5. HU Insertar Caso de Uso.....	38
Tabla 6. HU Mostrar listado de Casos de Uso.....	39
Tabla 7. HU Modificar Caso de Uso.....	39
Tabla 8. HU Modificar Caso de Uso.....	40
Tabla 9. HU Filtrar Listado de Casos de Uso.....	40
Tabla 10. HU Cambiar Estado de Caso de Uso.....	41
Tabla 11. HU Consultar CU.....	42
Tabla 12. HU Incluir Manual de Usuario.....	42
Tabla 13. HU Mostrar listado de Manuales de Usuario.....	43
Tabla 14. HU Modificar Manual de Usuario.....	43
Tabla 15. HU Eliminar Manual de Usuario.....	44
Tabla 16. HU Filtrar Listado de Manuales de Usuario.....	45
Tabla 17. HU Cambiar Estado de Manual de Usuario.....	45
Tabla 18. HU Consultar Manuales.....	46
Tabla 19. HU Generar Reporte.....	47
Tabla 20. Puntos de Estimación por Historia de Usuario.....	48
Tabla 21. Plan de Duración de las Iteraciones.....	50
Tabla 22. Plan de Entrega de las funcionalidades por Iteración.....	51
Tabla 23. Tarjeta CRC módulo Analista.....	54
Tabla 24. Tarjeta CRC módulo Calidad Interna.....	54

Tabla 25. Tarjeta CRC módulo Calidad UCI.....	54
Tabla 26. Tarjeta CRC módulo Desarrolladores.....	55
Tabla 27. Tarjeta CRC módulo Cliente VNZ.....	55
Tabla 28. Tarjeta CRC módulo Manual. ....	55
Tabla 29. Tarjeta CRC módulo Líder-Planificador. ....	56
Tabla 30. Clasificación de las NC .....	60
Tabla 31. Escala de evaluación de las preguntas del cuestionario.....	62

## Introducción

La gestión de proyecto es el proceso mediante el cual se planifica, dirige y controla el desarrollo de un sistema aceptable con un costo mínimo y dentro de un período de tiempo específico, siendo de vital importancia a quien asume la dirección del proyecto, pues se encarga de tomar las decisiones que ayudan a que este cumpla con los objetivos propuestos. Dentro del proceso de desarrollo de sistemas de software, la gestión de proyecto constituye una tarea de especial atención y sin embargo no ha ocupado el tiempo ni el esfuerzo que requiere para el éxito de un proyecto de software[1]. Producto de la importancia de este proceso, se impone la necesidad de utilizar como apoyo las nuevas tecnologías de la información. Para poder completar con éxito un proyecto, se necesita tener un control riguroso sobre el tiempo, las personas y las actividades que se desarrollan.

El proyecto de informatización del Cuerpo de Investigaciones Científicas, Penales y Criminalísticas de la República Bolivariana de Venezuela (CICPC), institución que garantiza la eficiencia en la investigación del delito, asegurando que el ejercicio de la acción penal conduzca a una sana administración de justicia, constituye un ejemplo de proyecto productivo que se realiza actualmente en la Universidad de las Ciencias Informáticas (UCI) con el objetivo de aumentar los ingresos económicos al país y apoyar el convenio de colaboración Cuba-Venezuela.

Para garantizar el desarrollo de un proyecto dentro de los plazos previstos es de trascendental importancia la planificación real del tiempo, costo y esfuerzo, de forma que le permita al gestor o líder del proyecto supervisar constantemente el avance del mismo, controlar el estado de las actividades planificadas, gestionar apropiadamente las situaciones no previstas y garantizar el cumplimiento de las metas productivas trazadas.

El proyecto CICPC se encuentra dividido en diferentes equipos de trabajo de acuerdo con las necesidades específicas del mismo, teniendo en cuenta como base, los roles fundamentales que plantea la metodología del Proceso Unificado de Desarrollo de Software (RUP). Para abarcar las funcionalidades que tendrá el nuevo sistema a desarrollar en el proyecto, se lleva a cabo la Captura de Requisitos donde se generan un grupo de artefactos a partir de la comprensión y modelamiento del negocio, este proceso culmina en una descripción detallada de cada una de las funcionalidades a implementar lo cual llamaremos Especificaciones de Casos de Uso, (en lo adelante ECU). CICPC es un proyecto de mucha complejidad, un gran dimensionamiento y cada etapa tiene como promedio un aproximado de 150 funcionalidades, por eso y para lograr el cumplimiento en tiempo de las metas previstas, es imprescindible controlar eficientemente el trabajo que se realiza durante la descripción de

las funcionalidades a implementar, revisión y aprobación de estas en Calidad Interna, Calidad UCI, aprobación por parte del cliente y escritura del Manual de Usuario, debido a que las ECU son artefactos muy importantes que constituyen la entrada a diversos flujos de trabajo. El proyecto CICPC no posee mecanismos eficientes que permitan la comunicación diaria del progreso y del estado actual de las actividades propias de los procesos mencionados anteriormente, el control se realiza de manera manual, mediante tablas y documentos enviados por correo electrónico. Esta situación dificulta la actualización diaria del avance y además se necesitan varias personas para llevar los controles, de esta forma, se ocasiona una descentralización tal de la información, que impide tener un dominio completo del estado general del proyecto.

Por lo tanto, existe un control deficiente de los procesos por los que atraviesa las ECU y los Manuales de Usuario, lo que conlleva a pérdida y descentralización de la información.

De acuerdo con la **situación problemática** descrita anteriormente, se plantea la siguiente interrogante como **problema a resolver**:

¿Cómo mejorar los procesos de organización y control de la información referente a las Especificaciones de Casos de Uso y Manuales de Usuario en el proyecto CICPC?

### **Objeto de Estudio**

Los procesos de organización y control de la información en un sistema de gestión.

### **Campo de Acción**

Control y seguimiento de las Especificaciones de Casos de Uso y Manuales de Usuario en el proyecto CICPC.

Como **objetivo general** se plantea lo siguiente: Elaborar una propuesta de software que contribuya al mejoramiento de los procesos de organización y control de los estados por los que pasan las Especificaciones de Casos de Uso y los Manuales de Usuario en el proyecto CICPC.

Los **objetivos específicos** que se persiguen son:

1. Realizar un diagnóstico sobre los procesos de organización y control de la información referente a las Especificaciones de Casos de Uso y Manuales de Usuario en un sistema de gestión.
2. Elaborar el marco teórico de la investigación.
3. Automatizar el proceso de seguimiento y control de los estados por los que pasan las Especificaciones de Casos de Uso y Manuales de Usuario en el proyecto CICPC.



4. Centralizar el control de la información referente a las ECU y Manuales de Usuario.
5. Validar la solución propuesta.

Como **idea a defender** se plantea lo siguiente: Con la automatización del proceso de control de las actividades referente a las ECU y los Manuales de Usuario, se contribuirá a la centralización de la información, así como a la reducción del tiempo en el cumplimiento de cada actividad planificada con estos artefactos.

Para lograr el objetivo propuesto se han trazado una serie de **tareas** a realizar:

- ✓ Estudio de las distintas fases, flujos y artefactos que genera la metodología Extreme Programming (XP).
- ✓ Realización de un estudio sobre la existencia de aplicaciones o soluciones similares a las características del sistema propuesto.
- ✓ Selección de las herramientas adecuadas para la implementación del sistema.
- ✓ Definición de las funcionalidades del sistema a desarrollar.
- ✓ Elaboración del diseño del sistema a desarrollar.
- ✓ Ejecución de la implementación de la aplicación.
- ✓ Ejecución de pruebas a la aplicación para la detección de errores.
- ✓ Despliegue de la aplicación.

Para el desarrollo de esta investigación se utilizaron métodos empíricos y teóricos, los cuales se detallan a continuación.

#### **Métodos Empíricos:**

- **Entrevista:** Para recopilar experiencias sobre cómo se organizan y controlan los procesos relacionados a la gestión de requisitos en un sistema de gestión.
- **Encuesta:** Para valorar el funcionamiento de la solución propuesta y evaluar los resultados aportados por la misma.

#### **Métodos Teóricos:**

- **Analítico–Sintético:** Para procesar toda la información enfocada hacia la investigación de los procesos de control de la información, permitiendo organizar y simplificar el análisis de todo el volumen de datos a recopilar en fracciones más factibles.

- **Histórico-Lógico:** Para conocer los antecedentes y tendencias actuales en los sistemas de gestión de proyectos, las plataformas y lenguajes de implementación que pueden ser utilizados en el desarrollo del sistema.

El presente trabajo se ha dividido en 5 capítulos. A continuación se resumen los temas abordados en cada uno de ellos.

### **Capítulo 1: Fundamentación Teórica.**

En este capítulo se analizan los sistemas similares vinculados al campo de acción. Se seleccionan las principales herramientas y tecnologías analizadas para la construcción de la solución propuesta haciendo un estudio comparativo de las mismas y se fundamenta la metodología de desarrollo de software utilizada para guiar el proceso de construcción de la solución.

### **Capítulo 2: Propuesta de Solución.**

En el capítulo 2 se describe el flujo actual de los procesos a automatizar. Se detallan las funcionalidades y características que tendrá la propuesta del sistema, lo que posibilitará un mayor entendimiento del mismo.

### **Capítulo 3: Exploración y Planificación.**

En este capítulo se detallan las historias de usuario definidas, que recogen todas las funcionalidades a implementar. Se muestran además otros artefactos propios de la metodología de desarrollo, que se generan en estas etapas.

### **Capítulo 4: Diseño e Implementación.**

En el capítulo 4 se hace referencia al diseño que tendrá el sistema a implementar y también se describen las tareas relacionadas a la fase de implementación.

### **Capítulo 5: Validación de la propuesta.**

En este capítulo se valida a través de pruebas de calidad, las funcionalidades del sistema construido. Se exponen los resultados obtenidos en las pruebas y se muestra la encuesta realizada a los usuarios del sistema para conocer el grado de adecuación de la herramienta al trabajo de los mismos.

## CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA.

### Introducción

En el presente capítulo se hace referencia a los elementos teóricos que fundamentan el desarrollo de la solución propuesta. Se analizan y presentan las tendencias y tecnologías actuales, permitiendo realizar una selección correcta de las mismas, que se adapte a las necesidades del sistema que se propone realizar. De manera que sea posible seleccionar el lenguaje de programación, el sistema gestor de base de datos, la metodología de desarrollo adecuada y las herramientas a utilizar en la construcción de la solución.

#### 1.1. Sistemas existentes vinculados al campo de acción:

##### 1.1.1. TRAC.

El TRAC es un sistema basado en web para la gestión y seguimiento de los proyectos de desarrollo de software[2]. Actualmente, en el proyecto CICPC se utiliza como una herramienta invaluable para los jefes de los diferentes equipos de trabajo y los integrantes de la dirección del proyecto, pues permite mantener una visión única y coherente de las principales iteraciones del desarrollo. El TRAC trabaja a través de tickets, los cuales sirven para asignar las tareas, los tickets son agrupados por milestone, los que definen de manera general las actividades de algún proceso específico.

Puede decirse que el nivel de avance del proyecto tiene 2 “vistas”, una horizontal expresada a través del cumplimiento de las tareas asignadas por equipos de trabajo: especificación de CU, realización de manuales de usuario, ayuda online, casos de prueba, análisis y diseño, implementación, pruebas y otras tareas que fácilmente pueden ser asignadas y controladas por el TRAC, debido a que pueden concretarse en tickets dirigidos específicamente a equipos de trabajo. Esta “vista horizontal” refleja el nivel de avance por iteraciones que se gestiona mediante milestones, y permite al líder y jefes de equipos trabajar en función del cumplimiento y cierre de cada iteración en el tiempo previsto.

Sin embargo, para la gestión eficiente del proyecto, es importante conocer además el avance basado en los CU completados por actividades, a lo cual se podría llamar la “vista vertical”, donde no sólo es necesario saber cuántos CU han sido especificados, liberados e implementados, sino también conocer en qué estado se encuentra el CU “x” y facilitar un medio de comunicación eficiente y centralizado que permita conocer el estado de cada CU y también notificar al equipo de trabajo “y” de cuáles son los CU listos para entrar a la siguiente iteración dentro del ciclo de vida del proyecto. Esta segunda vista es de

vital importancia y no es gestionada por el TRAC, pues no permite que el líder del proyecto pueda controlar y ver de manera global el estado de cada CU y Manual de Usuario en todos sus procesos, posibilitando que este tenga un control definido y organizado, de los flujos fundamentales del proyecto. No posibilita una comunicación diaria automatizada del progreso de cada actividad que se realiza en los diferentes equipos, a partir de algún estado que pueda tomar el ticket asignado a alguna persona y modificado por esta, los mismos no pueden pasar de un milestone a otro de forma automática de acuerdo con algún estado que estos puedan tomar. El TRAC sólo da la visión del desempeño personal en la realización de ciertas actividades que son asignadas a alguien específico. No da la medida exacta del avance en el desarrollo del proyecto de acuerdo a las actividades incluidas. Tampoco permite realizar reportes de la información mostrada y la misma es tan abundante que al líder del proyecto se le hace difícil conocer fácilmente el estado por el que se encuentra cada Manual de Usuario y Caso de Uso del proyecto. Por lo que se puede decir que no se llega a tener un dominio completo de la trazabilidad de estos componentes.

### 1.1.2. REDMINE.

El REDMINE es una herramienta web para la gestión de múltiples proyectos a la vez. Entre sus características se pueden destacar la gestión de tareas, incidencias, calendario de actividades, noticias, foros, wikis, gestión de ficheros y documentos, integración con Subversion<sup>1</sup>, control de tiempos, generación de informes, entre otros. Tiene similitudes con el TRAC en algunas de sus funcionalidades, además de tener una interfaz intuitiva, rápida y sencilla de usar. En este sistema existe un único concepto denominado petición que puede ser de tres tipos: tarea, soporte o incidencia. De esta forma todo se gestiona de igual manera, lo cual simplifica el trabajo pero a su vez puede convertirse en un volumen poco gestionable de tareas impidiendo tener una idea clara de la cantidad de las mismas y de qué representa su cumplimiento.

Sin embargo, partiendo del punto de que el objetivo principal de la utilización del REDMINE en la universidad y más específicamente en el proyecto CICPC, es controlar el trabajo de las personas implicadas en el desarrollo de un proyecto, a partir de la asignación diaria de tareas, se pueden encontrar un gran número de limitaciones que no dan respuesta a la necesidad de tener un control completo del estado actual del desarrollo del software realizado en el proyecto CICPC. Entre ellas se encuentra, que no brinda una información explícita del avance del proyecto a partir de cada flujo de actividades que se realiza dentro del mismo, pues no posibilita calcular de forma clara y evidente el

---

<sup>1</sup> Herramienta para el control de versiones.

avance del proyecto a partir del cumplimiento de las tareas asignadas. El porcentaje de cumplimiento de las tareas debe ser incluido por la persona a la que se le ha asignado y no la tarea en sí, la encargada de decir de acuerdo con el estado en que se encuentra, el porcentaje de avance de la actividad que engloba. Las peticiones solamente pueden ser asignadas a una persona, por lo que si se necesita que otra persona trabaje sobre ella, se debe incluir una nueva petición, lo cual muestra un sistema un tanto engorroso a la hora de enlazar peticiones, provocando un gran número de información, que pudiera simplificarse. Además, no permite de acuerdo con el estado de una tarea, que sea asignada automáticamente a otra persona, que le dará continuidad al trabajo realizado con la misma. Otra de las deficiencias es el hecho de que una vez creada una petición, su nombre y descripción no se pueden modificar, tan sólo ir añadiendo nuevas descripciones, por lo que si en algún momento se necesita cambiar algún dato de la petición, las modificaciones no serán vistas a primera instancia sino que habrá que acceder a ver los detalles de la misma para conocer los cambios o sencillamente crear una nueva. Por lo que se puede decir que, con la utilización del REDMINE en el proyecto, aún se sigue necesitando de un gran número de personas que actualicen el trabajo realizado. El gran cúmulo de tareas no permite tener una visión clara del avance del proyecto de acuerdo con el estado que tengan las mismas y no existe forma de que automáticamente una petición se reasigne por sí sola, de acuerdo con su estado, a otra persona, encargada de darle seguimiento y continuidad a la actividad que la petición describe.

### 1.1.3. E-GROUPWARE.

E-GroupWare es un conjunto de herramientas que permiten la coordinación de grandes grupos de personas que trabajan en una misma organización. La finalidad de esta familia de aplicaciones es la de facilitar la distribución de información y la planificación de reuniones y eventos colectivos, así como de la gestión de los recursos necesarios para los mismos[3]. Es una solución de trabajo en grupo vía web de código abierto. Está escrita en PHP utilizando bases de datos, como PostgreSQL o MySQL. Entre las herramientas y facilidades que ofrece al usuario se encuentran la agenda, el calendario, notas, almacenamiento de archivos compartidos, control de proyectos, entre otros.

Sin embargo, esta herramienta está más enfocada a la planificación del tiempo y eventos de las personas implicadas, que al progreso global del proyecto a partir de la finalización de las tareas. Las tareas generadas en los proyectos no tienen fecha de inicio y su estado por defecto es "nonactive". De esta manera, las personas encargadas de realizarlas no las ven como tareas pendientes en el calendario hasta que un distribuidor valida que es el momento de comenzar con ellas. La finalización

de una tarea no aporta información del avance del proyecto de forma automática. Por tanto, es necesario que la tarea sea controlada por una persona para que traslade la información al porcentaje de desarrollo y progreso del proyecto. Las tareas son gestionadas dentro de sub-proyectos en el sistema, pero no existe la forma de que a partir de algún estado que pueda tomar la tarea, esta pase automáticamente a otro sub-proyecto, o que sea reasignada a otra persona para que trabaje sobre ella. Se puede decir que el E-GroupWare es mucho más eficiente para la gestión del tiempo, recursos y eventos dentro de una empresa, que para contemplar el avance de un proyecto a partir de las tareas que son asignadas a sus miembros.

El estudio realizado de las distintas herramientas de gestión de proyectos utilizadas actualmente por los miembros de los diferentes equipos de trabajo del proyecto CICPC, demostró la necesidad de la creación de un nuevo mecanismo para controlar el avance del mismo a partir de los estados que pudieran tomar los Casos de Uso, durante el proceso de desarrollo del software. Se pudo probar que ninguna de las herramientas mencionadas anteriormente, a pesar de ser muy potentes y brindar muchos servicios, se adaptan a las necesidades reales del líder del proyecto de forma que le facilite establecer porcentajes reales del progreso del desarrollo, a partir de los reportes que se obtengan, lo que posibilitaría reajustar cronogramas y mitigar posibles riesgos. Sólo controlan el trabajo realizado por cada persona individualmente y no la productividad y el avance del proyecto a partir de la finalización de cada actividad. No existe forma de que las tareas asignadas, una vez concluidas pasen de forma automática a otro equipo encargado de darle seguimiento. Siempre se hace necesario incluir una nueva tarea y asignársela al responsable de la próxima actividad a realizar con ella, obteniéndose un cúmulo innecesario de peticiones de trabajo. Teniendo en cuenta todas estas limitaciones puntuales, se considera necesario, implementar un nuevo sistema que cubra y ofrezca respuestas a esas necesidades específicas que aún existen en el proyecto, lo que posibilitará mayor eficiencia a la hora de controlar el progreso del proyecto partiendo de las actividades completadas con los Casos de Uso y Manuales de Usuario.

### **1.2. Tendencias y tecnologías actuales.**

En el presente epígrafe se exponen algunas de las tecnologías y tendencias actuales que pueden ser utilizadas para solucionar los problemas antes enunciados de forma eficiente, según las necesidades existentes.

### 1.2.1. Aplicaciones Web.

En la ingeniería de software se denomina aplicación web a aquellas aplicaciones que los usuarios pueden utilizar accediendo a un servidor web a través de Internet o de una Intranet utilizando un navegador. Es una aplicación de software que se codifica en un lenguaje soportado por los navegadores web HTML (Hypertext Markup Language), JavaScript, Java, asp.net,php, entre otros, en la que se confía la ejecución al navegador.

Las aplicaciones web son populares debido a lo práctico del navegador web como cliente ligero, así como a la facilidad para actualizar y mantener las aplicaciones web sin distribuir e instalar un software a miles de usuarios potenciales. Existen aplicaciones como los webmails, wikis, weblogs, tiendas en línea y la propia Wikipedia que son ejemplos bien conocidos de aplicaciones web[4].

### 1.2.2. Servidores Web.

Un servidor web es un programa que implementa el protocolo Hypertext Transfer Protocol (HTTP, por sus siglas en inglés). Este protocolo pertenece a la capa de aplicación del modelo Open System Interconnection (OSI) y está diseñado para transferir hipertextos, páginas web o páginas HTML, textos complejos con enlaces, figuras, formularios, botones y objetos incrustados como animaciones o reproductores de música. De acuerdo con una encuesta realizada por la empresa inglesa Netcraft, en septiembre del 2009, los servidores web más usados en la actualidad son Internet Information Services (IIS) y Apache[5].

#### 1.2.2.1. Internet Information Services (IIS).

Internet Information Services, es una serie de servicios para los ordenadores que funcionan con Windows. Originalmente era parte del Option Pack para Windows NT, luego fue integrado en otros sistemas operativos de Microsoft destinados a ofrecer servicios, como Windows 2000 o Windows Server 2003. Windows XP Profesional incluye una versión limitada de IIS.

Este servicio convierte a un ordenador en un servidor de Internet o Intranet, es decir, en las computadoras que tienen este servicio instalado se pueden publicar páginas web locales o de forma remota.

Los servicios de IIS proporcionan las herramientas y funciones necesarias para administrar de forma sencilla un servidor web seguro. El servidor web se basa en varios módulos que le dan capacidad para procesar distintos tipos de páginas, por ejemplo Microsoft incluye los de Active Server Pages (ASP).

También pueden ser incluidos los de otros fabricantes, como PHP o Perl. IIS es reconocido por presentar entre sus ventajas:

- Confiabilidad y escalabilidad.
- Seguridad y capacidad de administración.
- Buen desarrollo y compatibilidad internacional.

### 1.2.2.2. Apache.

Apache es un servidor web que permite el acceso a páginas contenidas en un ordenador. Actualmente es uno de los servidores web más usados. Tiene una gran simplicidad, posee tres ficheros en texto plano para establecer la configuración del servidor. Admite servidores virtuales, posee la capacidad para registrar las acciones que realiza el usuario mediante cookies. Permite el trabajo con Common Gateway Interface (CGI, por sus siglas en inglés).

El servidor Apache está dividido en módulos, porciones de código que hacen referencia a funcionalidades del mismo. Esta modularidad es con la intención de que la configuración de cada módulo, se hace configurando las directivas contenidas dentro del módulo. Los módulos se dividen en tres categorías:

- **Módulos Base:** Módulo con las funciones básicas de Apache.
- **Módulos Multiproceso:** Son los responsables de la unión con los puertos de la máquina, aceptando las peticiones y enviando a los hijos a atender a las peticiones.
- **Módulos Adicionales:** Cualquier otro módulo que le añada una funcionalidad al servidor.

En el Módulo Base se encuentran las funciones elementales, por lo que es necesario un Módulo Multiproceso para manejar las peticiones. Existen varios Módulos Multiprocesos para cada una de las plataformas donde se puede ejecutar el Apache, optimizando así el rendimiento. Mediante Módulos Adicionales, que pueden ser cargados, se puede hacer uso del resto de las funcionalidades. Teniendo en cuenta las características anteriores se decidió utilizar Apache como servidor web, para la implantación de la aplicación que se propone en esta investigación.

### 1.2.3. Lenguajes de Programación.

Un lenguaje de programación es el medio que se utiliza para hacerle entender a las máquinas, particularmente a las computadoras, las acciones que queremos que estas realicen. Mediante este, el



programador especifica la forma en que los datos serán procesados, almacenados y transmitidos, además de establecer el comportamiento del programa, dadas diferentes condiciones[6].

### 1.2.3.1. Java.

Es un lenguaje orientado a objetos e independiente de la plataforma, el cual se caracteriza por:

- Ser robusto.
- Gestionar memoria automáticamente.
- Mecanismos de seguridad incorporados.
- Herramientas de documentación incorporadas.

Java también aumenta el contenido multimedia de los sitios, brindando animaciones fluidas, mejora en los gráficos, sonido y video.

Sin embargo, como todo lenguaje de programación tiene sus limitaciones, entre las cuales se pueden encontrar:

- Es mucho más pesado y lento en cuanto a su rendimiento que otros lenguajes.
- Tiene un gran coste de formación de los desarrolladores que lo utilizan. Java es mucho más complicado de aprender que otros lenguajes como PHP, C, entre otros.
- Se necesita tener instalada una JVM (Máquina Virtual de Java, por sus siglas en inglés), para correr aplicaciones programadas en este lenguaje.
- Tiene un gran desperdicio de memoria, pues trabaja con la memoria usada y no la necesitada realmente.

### 1.2.3.2. Hypertext Preprocessor (PHP).

Es una tecnología del lado del servidor, gratuito e independiente de plataforma, rápido, con una gran librería de funciones y mucha documentación[7].

PHP se escribe dentro del código HTML, lo que lo hace realmente fácil de utilizar, al igual que ocurre con ASP de Microsoft, pero con algunas ventajas como su gratuidad, independencia de plataforma, rapidez y seguridad. Se puede descargar a través de la página principal de PHP [www.php.net](http://www.php.net) y de manera gratuita un módulo que hace que cualquier servidor web comprenda los scripts realizados en este lenguaje. Puede ser desplegado en la mayoría de los servidores web y en casi todos los sistemas

operativos y plataformas sin costo alguno. Es un lenguaje de alta potencia y fácil de usar, e incluye la programación orientada a objetos. Tiene múltiples ventajas entre las cuales se pueden encontrar:

- Es muy fácil de aprender.
- Se caracteriza por ser un lenguaje muy rápido.
- Soporta la orientación a objeto (clases y herencia).
- Es un lenguaje multiplataforma: Linux, Windows, entre otros.
- Posee una amplia documentación.
- Presenta una biblioteca nativa de funciones muy extensa.
- Tiene soporte para conectarse a una gran variedad de base de datos como: MySQL, PostgreSQL, SQL y Oracle.
- Es libre, por lo que se presenta como una alternativa de fácil acceso para todos.

### **1.2.3.3. Fundamentación del lenguaje de programación a utilizar.**

Teniendo en cuenta las características de los lenguajes antes mencionados, se decide usar PHP como lenguaje de programación para desarrollar el sistema propuesto. Es uno de los lenguajes más usados en la actualidad por su rapidez y la gran documentación que posee. Consume pocos recursos, es fácil de aprender, comprender y se adapta perfectamente a las necesidades del sistema que se quiere desarrollar, por lo que se considera la mejor opción.

### **1.2.4. Sistema Gestores de Base de Datos.**

Los sistemas de gestión de base de datos o SGBD (en inglés: Database Management System, abreviado DBMS) son un tipo de software muy específico, dedicado a servir de interfaz entre la base de datos, el usuario y las aplicaciones que la utilizan[8]. El propósito general de los sistemas de gestión de base de datos es manejar de manera clara, sencilla y ordenada un conjunto de datos que posteriormente se convertirán en información relevante.

El criterio principal que se utiliza para clasificar los SGBD es el modelo lógico en que se basan. Los modelos lógicos empleados con mayor frecuencia en los SGBD comerciales actuales son: el relacional, el de red y el jerárquico. Algunos SGBD más modernos se basan en modelos orientados a objetos.

### 1.2.4.1. MySQL.

Es un sistema de administración de base de datos utilizado para web. MySQL utiliza el lenguaje SQL (Structure Query Language, por sus siglas en inglés), un estándar que utilizan también otros muchos sistemas de este tipo. Desarrollado por MySQL AB como software libre en un esquema de licenciamiento dual. MySQL es muy utilizado en aplicaciones web como MediaWiki o Drupal, en múltiples plataformas, como Linux, Windows y Solaris y por herramientas de seguimiento de errores como Bugzilla. Su popularidad está muy ligada a PHP, que a menudo aparece en combinación con MySQL. Su principal objetivo de diseño es la velocidad, sacrificándose algunas características esenciales en sistemas más "serios" con este fin. Otra característica importante es que consume muy pocos recursos, tanto de CPU como de memoria. Entre sus ventajas se pueden encontrar las siguientes:

- Gran rendimiento.
- Mayor velocidad tanto al conectar con el servidor como al servir consultas (incluso más rápida que PostgreSQL).
- Mejor integración con PHP.
- No hay límites en el tamaño de los registros.
- Mejor control de acceso, en el sentido de qué usuarios tienen acceso a qué tablas y con qué permisos.
- MySQL se comporta mejor que Postgre a la hora de modificar o añadir campos a una tabla.

### 1.2.4.2. PostgreSQL.

PostgreSQL es el Sistema de Gestión de Bases de Datos Objeto-Relacionales (ORDBMS) de código abierto más avanzado del mundo. Postgre intenta ser un sistema de bases de datos de un gran nivel a la altura de Oracle, Sybase o Interbase. Entre las ventajas de su uso se encuentran:

- Por su arquitectura de diseño, escala muy bien al aumentar el número de CPU y la cantidad de memoria RAM.
- Soporta transacciones y desde la versión 7.0, claves ajenas (con comprobaciones de integridad referencial).
- Tiene mejor soporte para triggers y procedimientos en el servidor.

### **Inconvenientes:**

- Consume muchos más recursos y carga más el sistema.
- Límite del tamaño de cada fila de las tablas a 8k (se puede ampliar a 32k recompilando, pero con un costo añadido en el rendimiento).
- Es de 2 a 3 veces más lenta que MySQL y más si se trata de su uso en aplicaciones web.
- Tiene menos funciones en PHP.

### **1.2.4.3. Fundamentación del Sistema de Base de Datos a utilizar.**

El estudio de los sistemas gestores de base de datos más usados en la actualidad ha demostrado que MySQL es la mejor opción y la más adecuada a las necesidades del sistema a implementar. MySQL junto con Apache y PHP forman un buen equipo para el desarrollo de aplicaciones web. Es, probablemente, el gestor más usado en el mundo, debido a su gran rapidez y facilidad de uso. Actualmente existen infinidad de librerías y otras herramientas que permiten su uso a través de una gran cantidad de lenguajes de programación, conjuntamente con su fácil instalación y configuración. Teniendo en cuenta que el sistema propuesto, donde se pretende usar MySQL, será una aplicación pequeña a la que no tendrán acceso un gran número de usuarios, este se convierte en el gestor de base de datos más recomendado. También se tiene un conocimiento previo sobre este gestor, lo que permitirá al equipo de desarrollo optimizar el tiempo durante la construcción del software. Aunque las versiones más actualizadas de MySQL son de software propietario, debido a la amplia comunidad de usuarios que lo utilizan, muchos desarrolladores han seguido trabajando en el perfeccionamiento de su versión libre, por lo que no se ha olvidado su mantenimiento.

Las principales características de este gestor de bases de datos que se tuvieron en cuenta para su selección fueron:

- Soporta gran cantidad de tipos de datos para las columnas.
- Tiene mejor documentación que PostgreSQL y se ha orientado más a facilitarle la vida al desarrollador proporcionando mejores herramientas de administración.
- Dispone de API's en gran cantidad de lenguajes (C, C++, Java, PHP, entre otras).
- Gran portabilidad entre sistemas.
- Gestión de usuarios y contraseña, manteniendo un buen nivel de seguridad en los datos.

- Alta velocidad a la hora de realizar las operaciones, por lo que es uno de los gestores que ofrece mayor rendimiento.
- Bajo consumo de los recursos de la máquina.
- Gran facilidad de configuración e instalación.
- Presenta una probabilidad muy reducida de corromper los datos.

### 1.2.5. Metodología de Desarrollo de Software.

Una metodología de desarrollo de software es un conjunto de pasos, procedimientos y técnicas que deben seguirse para desarrollar un software. Las metodologías tienen como propósito el desarrollo de un software eficaz y eficiente que reúna los requisitos del cliente.

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 especial hincapié en la planificación total de todo el trabajo a realizar y una vez que está todo detallado, comienza el ciclo de desarrollo del producto de software.

**Metodologías Ágiles:** 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.

#### 1.2.5.1. Rational Unified Process (RUP).

RUP 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[9]. Es uno de los procesos más generales que existen actualmente, pues en realidad está pensado para adaptarse a cualquier proyecto y no tan sólo de software.

Se caracteriza por ser **dirigido por casos de uso**, donde los casos de uso definen lo que el usuario desea a partir de la captura de requisitos y la modelación del negocio. Es **centrado en la arquitectura**, característica que brinda una visión completa del sistema. Y por último es **iterativo e incremental**, donde cada fase se desarrolla en iteraciones, de forma tal que se pueda dividir en pequeños proyectos

mejorando su comprensión y desarrollo. Un proyecto realizado siguiendo RUP se divide en cuatro fases:

- Inicio.
- Elaboración.
- Construcción.
- Transición.

En cada fase se ejecutarán una o varias iteraciones de tamaño variable según el proyecto, y dentro de cada una de ellas se seguirá un modelo de cascada para los flujos de trabajo. RUP define nueve disciplinas a realizar en cada ciclo del proyecto, donde seis de ellas son flujos de trabajo básicos (Modelamiento del Negocio, Requerimientos, Análisis y Diseño, Implementación, Prueba e Instalación) y las otras disciplinas de soporte (Administración del Proyecto, Administración de Configuración y Cambios y Ambiente).

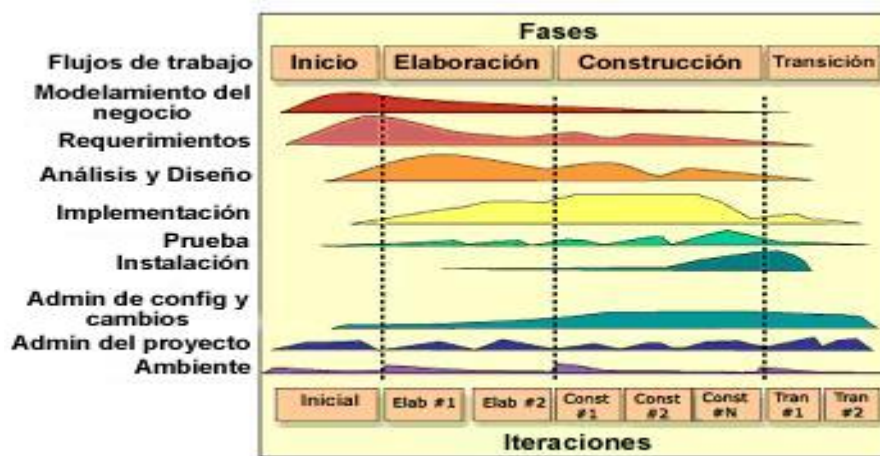


Figura 1. Flujos de Trabajo de RUP.

RUP está pensado para proyectos y equipos grandes, en cuanto a tamaño y duración. Es un proceso pesado, muy basado en la documentación, en la que los cambios no son deseados. Los artefactos se presentan al cliente al final de cada fase y sólo después de que estos son aceptados por el cliente se pasará a la siguiente fase. En resumen, es una metodología más apropiada para proyectos de gran envergadura dado que requiere un equipo de trabajo capaz de administrar un proceso complejo en varias etapas.

### 1.2.5.2. Extreme Programming (XP).

XP es la primera metodología ágil y la que le dio conciencia al movimiento actual de metodologías ágiles. De la mano de Kent Beck, XP ha conformado un extenso grupo de seguidores en todo el mundo, disparando una gran cantidad de libros a los que comenzó el mismo Beck[10].

XP como metodología ágil está centrada en potenciar las relaciones interpersonales como clave para el éxito en el desarrollo de un software, promoviendo el trabajo en equipo, preocupándose por el aprendizaje de los desarrolladores y propiciando un buen clima de trabajo[11]. Se basa en la realimentación continua entre el cliente y el equipo de desarrollo, comunicación fluida entre todos los participantes, simplicidad en las soluciones implementadas y coraje para enfrentar los cambios. Se define como especialmente adecuada para proyectos con requisitos imprecisos y muy cambiantes, donde existe un alto riesgo técnico[12].

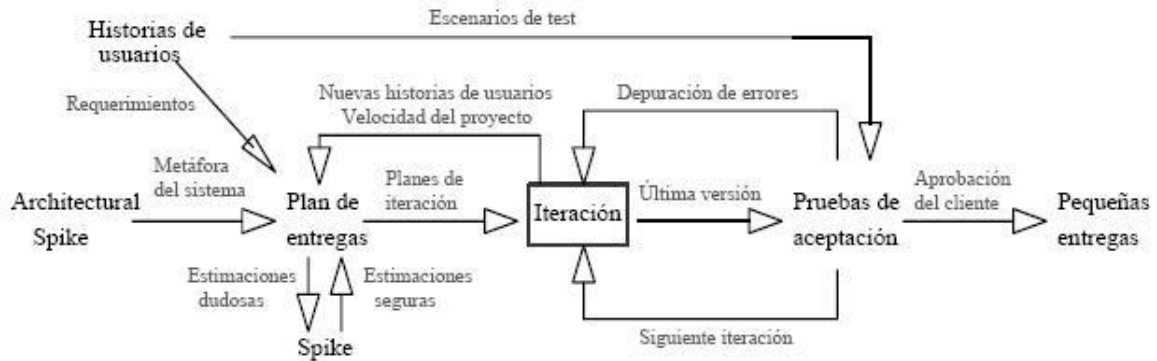
En XP aparece un nuevo concepto llamado "*Historia de Usuario*", las cuales tienen el mismo propósito que los casos de uso. Se trata de una lista de características que el cliente necesita que existan en el producto final. Las historias de usuario las escriben los propios clientes, tal y como ven ellos las necesidades del sistema. Son similares al empleo de escenarios, con la excepción de que no se limitan a la descripción de la interfaz de usuario y solamente proporcionarán los detalles sobre la estimación del riesgo y cuánto tiempo conllevará la implementación de las mismas. También conducirán el proceso de las pruebas de aceptación.

La metodología XP consta de cuatro fases fundamentales:

- Planificación.
- Diseño.
- Implementación.
- Pruebas.



## Proyecto en XP



Spike = Pequeño programa que explora posibles soluciones potenciales

Figura 2: Fases de un proyecto en XP.

En la fase de Planificación, las actividades fundamentales están dirigidas a definir las historias de usuario con el cliente y una vez concluido este proceso se precisa el *Plan de Entrega*, donde se indican las historias de usuario que se implementarán en cada versión del programa y las fechas en las que se publicarán estas versiones. Todo el proyecto se realizará en iteraciones y se planificarán reuniones diarias en conjunto con el cliente para exponer el avance y las dificultades que se tengan en el proyecto. La fase de Diseño se basa en definir el diseño que tendrá cada versión que se le entregue al cliente, centrándose siempre en diseños simples y teniendo cuidado de no añadir funcionalidades que no sean importantes para la entrega de cada iteración. La fase de Implementación es la programación pura del sistema, donde se opta por la programación en pareja, ya que permite un código más eficiente y con una gran calidad. Finalmente, la fase de Pruebas es uno de los pilares más importantes, pues es en ese momento, donde se comprueba el funcionamiento de los códigos que se hayan implementado con las distintas historias de usuario y si, realmente, cumplen su cometido.



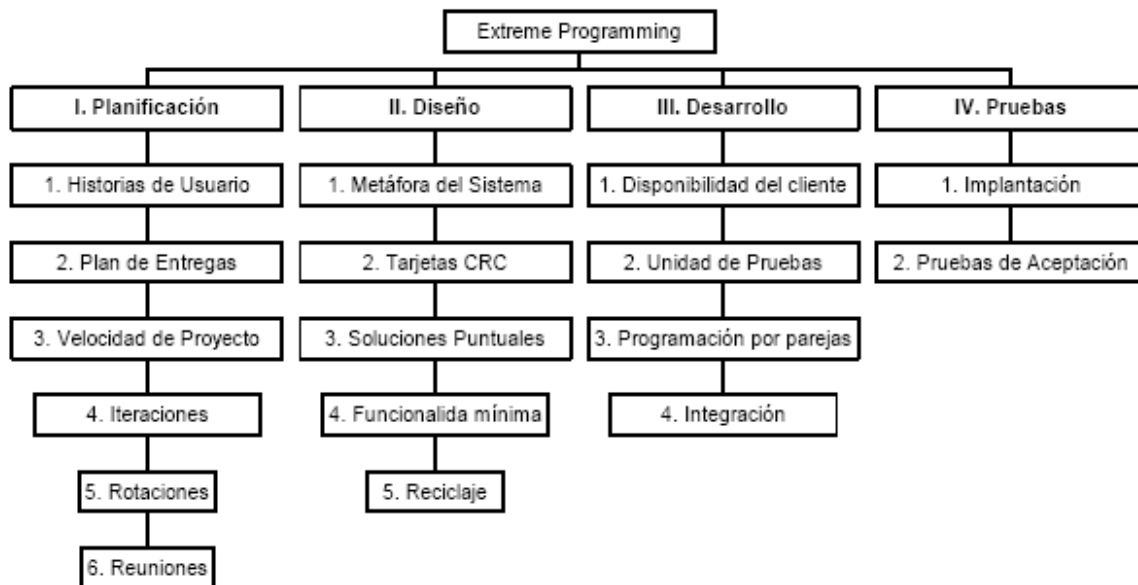


Figura 3. Etapas de XP.

El ciclo de vida de un proyecto, que utiliza XP como metodología de desarrollo, está compuesto por 6 etapas.

- 1- **Exploración:** Donde los clientes plantean a grandes rasgos las historias de usuario que son de interés para la primera entrega del producto. Al mismo tiempo, el equipo de desarrollo se familiariza con las herramientas, tecnologías y prácticas que se utilizarán en el proyecto. Se prueba la tecnología y se exploran las posibilidades de la arquitectura del sistema, construyendo un prototipo.
- 2- **Planificación de la Entrega (Release):** En esta fase el cliente establece la prioridad de cada historia de usuario y correspondientemente, los programadores realizan una estimación del esfuerzo necesario para cada una de ellas. Se toman acuerdos sobre el contenido de la primera entrega y se determina un cronograma en conjunto con el cliente.
- 3- **Iteraciones:** Esta fase incluye varias iteraciones sobre el sistema antes de ser entregado.
- 4- **Producción:** Esta fase consta de un Plan de Entrega, que está compuesto por iteraciones de no más de seis semanas.
- 5- **Mantenimiento:** Mientras la primera versión se encuentra en producción, el proyecto XP debe mantener el sistema en funcionamiento al mismo tiempo que desarrolla nuevas iteraciones. La fase de mantenimiento puede requerir un nuevo personal dentro del equipo y cambios en su estructura.
- 6- **Muerte del Proyecto:** Es cuando el cliente no tiene más historias de usuario para ser incluidas en el sistema. Se genera la documentación final del sistema y no se realizan más cambios en la

arquitectura. La muerte del proyecto también ocurre cuando el sistema no genera los beneficios esperados por el cliente o cuando no hay presupuesto para mantenerlo.

### 1.2.5.3. Fundamentación de la metodología de desarrollo a utilizar.

Para desarrollar la propuesta que presenta esta investigación, se ha decidido utilizar como metodología de desarrollo, XP, pues resulta ser la más idónea, teniendo en cuenta las características del sistema que se quiere implementar, las cuales se mencionan a continuación:

- **Es un proyecto pequeño:** El proyecto que se desarrollará no tiene un tiempo de desarrollo extenso.
- **Mayor claridad en la comprensión de los requisitos:** Con la utilización de XP las funcionalidades y características del sistema quedarán mejor definidas, pues es el propio cliente el encargado de escribirlas a través de las historias de usuario.
- **El riesgo del desarrollo es alto.** Debido al poco tiempo de desarrollo del proyecto y a los cambios continuos de funcionalidades que pudieran ocurrir, XP ofrece un diseño que permite mitigar los riesgos que pudieran ocurrir en este tipo de proyecto.
- **Poca disponibilidad de personal:** El sistema está diseñado para que se desarrolle por dos personas, por lo que no da lugar a la definición de un gran número de roles, ni a la especialización de los mismos.
- **El cliente forma parte del equipo de desarrollo:** Utilizando XP se puede lograr una mayor retroalimentación con el cliente, posibilitando que sus necesidades sean satisfechas totalmente.
- **Pequeñas entregas:** Con XP, el proyecto va creciendo poco a poco hasta alcanzar un producto final. El proyecto es dividido en funcionalidades más pequeñas de tal manera que se puede hacer entrega funcional de partes del producto mientras se avanza con el resto.
- **Realización continua de pruebas:** La realización de Pruebas de Aceptación con el cliente al final de cada iteración, implica desarrollar y entregar partes funcionales en correspondencia con los requisitos deseados por el mismo.

### 1.2.6. Sistema de Gestión de Contenidos.

Los sistemas de gestión de contenidos (Content Management Systems o CMS) son herramientas que permiten crear y mantener un sistema web con facilidad, encargándose de los trabajos más tediosos

que hasta ahora ocupaban el tiempo de los administradores de la web[13]. Es un tipo de software que se utiliza principalmente para facilitar la gestión de web, ya sea en Internet o en una Intranet y por eso también son conocidos como gestores de contenido web (Web Content Management o WCM, en inglés). Las ventajas del uso de los CMS radica principalmente en el control que se puede lograr con estos, pues son sistemas que controlan que sólo las personas autorizadas tengan acceso al mantenimiento de la aplicación web, lo cual mejora tanto la calidad, como la eficiencia de la misma. La utilización de CMS se ha hecho común para la mayoría de los desarrolladores de aplicaciones web, lo que ha llevado a la creación de un gran número de CMS. Entre los más usados a nivel mundial, para el desarrollo de aplicaciones web, podemos encontrar a Drupal y a Joomla. Estos presentan características comunes entre sí, e incluyen de forma independiente funcionalidades que los caracteriza y los diferencia a su vez.

### **1.2.6.1. CMS Drupal.**

Drupal es un sistema de gestión de contenidos modular y muy configurable. Es un programa de código abierto, con licencia GNU/GPL, escrito en PHP, desarrollado y mantenido por una activa comunidad de usuarios. Se destaca por la calidad de su código y de las páginas generadas, el respeto de los estándares de la web y un énfasis especial en la usabilidad y consistencia de todo el sistema. Sus principales características son su alto rendimiento, escalabilidad y una flexibilidad de diseño significativa. Drupal cuenta con una buena gestión de usuarios, agrupados por roles, lo que hace más fácil la asignación de permisos a cada uno. Está compuesto por una infraestructura base y ha sido diseñado con una arquitectura modular. Gracias a esta arquitectura se pueden agregar nuevas funciones a módulos ya existentes y crear nuevos módulos sin afectar los ya creados. Estos módulos son los encargados del funcionamiento del sistema y separan la interfaz gráfica de la información[14].

### **1.2.6.2. CMS Joomla.**

Joomla es un sistema administrador de contenidos que les permite a los usuarios publicar fácilmente contenidos en Internet. Está escrito en lenguaje PHP y SQL. Utiliza bases de datos relacionales, más específicamente MySQL. Tanto PHP como MySQL son programas de libre distribución y uso. La administración de Joomla está enteramente basada en la gestión online de contenidos ya que todas las acciones que realizan los administradores de los sitios Joomla, ya sea para modificar, agregar o eliminar contenidos, se realiza exclusivamente mediante un navegador web conectado a Internet, es decir, a través del protocolo HTTP. Es de uso gratuito, de libre distribución y de código abierto. El mismo se usa y distribuye bajo licencia pública general (GNU/GLP).

### **1.2.6.3. Fundamentación de la selección del CMS a utilizar.**

Teniendo en cuenta las características de los CMS descritos anteriormente, se ha decidido utilizar el CMS Drupal como plataforma de publicación de la aplicación propuesta. Drupal brinda un gran número de facilidades, que lo hacen adecuado para realizar muchos tipos de sitios web. Entre esas facilidades se puede destacar su rápido y bajo consumo del servidor. Además, ofrece muchos recursos y módulos incluidos muy sencillos de administrar. Drupal tiene un buen diseño y un código de muy buena calidad, que lo hace fácilmente extensible. Su usabilidad y accesibilidad son excelentes en comparación con otros CMS. Es abierto a PHP permitiendo que en cualquier tipo de contenido se pueda agregar código PHP en dependencia de las necesidades, además, genera un código HTML extremadamente limpio y bien estructurado. Permite tener aplicaciones multilingüaje, así como hacer cualquier diseño deseado. Tiene un control total para los administradores, desde permisos para usuario y módulo hasta el manejo del panel de control.

Por último se debe señalar que cuenta con un soporte de comunidad muy amplio, donde sus usuarios colaboran en un fin común y esa sea quizás la principal ventaja de Drupal frente a Joomla, pues Drupal tiene una comunidad muy buena, Drupal.org, que es un punto de encuentro entre sus seguidores donde cualquier duda está documentada y explicada.

### **1.2.7. Otros lenguajes utilizados.**

#### **1.2.7.1. Hojas de estilo en cascada (CSS).**

Las hojas de estilo en cascada (en inglés Cascading Style Sheets o CSS) es un lenguaje usado para definir la presentación de un documento estructurado escrito en HTML o XML. Fue desarrollado por el W3C (World Wide Web Consortium, por sus siglas en inglés) para permitir la separación de los contenidos HTML, de la presentación del documento, con las hojas de estilo, incluyendo elementos tales como los colores, fondos, márgenes, bordes, tipos de letra, modificando la apariencia de una página web de una forma más sencilla, permitiendo a los desarrolladores controlar el estilo y formato de sus documentos[15]. Con CSS los navegadores permiten a los usuarios especificar su propia hoja de estilo local que será aplicada a un sitio web, con lo que aumenta considerablemente la accesibilidad. Por ejemplo, personas con deficiencias visuales pueden configurar su propia hoja de estilo para aumentar el tamaño del texto o remarcar más los enlaces. Además, posibilita tener un control centralizado de la presentación de un sitio web completo con lo que se agiliza de forma considerable la actualización del mismo[16].

### 1.2.7.2. **Java Script.**

JavaScript es un lenguaje interpretado, orientado a las páginas web, con una sintaxis semejante a la del lenguaje Java. Se utiliza en páginas web HTML, para realizar tareas y operaciones en el marco de la aplicación cliente[16]. Está destinado a la implementación de programas pequeños, es ideal para tareas repetitivas y de control de eventos. JavaScript no requiere un tiempo de compilación, los scripts se pueden desarrollar en un período de tiempo relativamente corto. A esto podemos añadirle las características de interfaz como: cuadro de diálogo, formularios y otros elementos GUI (Interfaz Gráfico de Usuario), que son gestionados por el navegador y por el código HTML. Es independiente de la plataforma, hardware o sistema operativo, al igual que los programas que utilicen este lenguaje, siempre y cuando exista un navegador con soporte JavaScript para la plataforma en cuestión[17].

### 1.2.8. **Otras herramientas de apoyo para el desarrollo.**

#### 1.2.8.1. **Zend Studio.**

Zend Studio o Zend Development Environment es un completo entorno de desarrollo integrado para el lenguaje de programación PHP. Se trata de un programa de la casa Zend, impulsores de la tecnología de servidor PHP, orientada a desarrollar aplicaciones web.

Zend Studio consta de dos partes en las que se dividen las funcionalidades del cliente y las del servidor. Ambas se instalan por separado, la del cliente contiene la interfaz de edición y la ayuda. Permite además, hacer depuraciones simples de scripts, aunque para disfrutar de toda la potencia de la herramienta de depuración habrá que disponer de la parte del servidor, que instala Apache y el módulo PHP, o en caso de que estén instalados, los configura para trabajar juntos en depuración.

#### 1.2.8.2. **Aptana.**

Aptana Studio es una solución completa de entorno de desarrollo para aumentar de forma notable la productividad, trabajo en grupo, conexión con control de versiones, depuración, todo lo que cualquier programador busca en este tipo de entornos. Destaca como un entorno para la programación de aplicaciones web dinámicas y en especial con soporte para AJAX, además de dar un completo e interesante soporte para plataformas emergentes como son Adobe AIR e iPhone/iPod Touch.

#### 1.2.8.3. **Dreamweaver.**

Adobe Dreamweaver es una aplicación en forma de estudio destinada a la construcción y edición de sitios y aplicaciones web basadas en estándares. Creado inicialmente por Macromedia (actualmente

producido por Adobe Systems), es el programa de este tipo más utilizado en el sector del diseño y la programación web, por sus funcionalidades, su integración con otras herramientas como Adobe Flash y, recientemente, por su soporte de los estándares del World Wide Web Consortium. Tiene soporte tanto para edición de imágenes como para animación a través de su integración con otras herramientas. Dreamweaver permite al usuario utilizar la mayoría de los navegadores web instalados en su ordenador para pre-visualizar las páginas web. También dispone de herramientas de administración de sitios dirigidas a principiantes como por ejemplo, la habilidad de encontrar y reemplazar líneas de texto y código por cualquier tipo de parámetro especificado, hasta el sitio web completo.

#### **1.2.8.4. NuSphere.**

NuSphere es un editor de programación especializado en PHP e incluye resaltado de sintaxis para varios lenguajes de programación como PHP, Perl, Javascript, SQL y HTML. Puede también auto-completar el código, insertar trozos de código personalizados que se utilizan con frecuencia. Dispone de resaltado de errores y ayuda a debuguear<sup>2</sup> las aplicaciones.

### **Conclusiones.**

En este capítulo se realizó un estudio de las tendencias y tecnologías actuales para el desarrollo de aplicaciones web, donde se pudieron apreciar las principales características de las mismas teniendo en cuenta para su selección, el conocimiento previo sobre estas tecnologías, con el objetivo de minimizar el tiempo de desarrollo del sistema a realizar.

---

<sup>2</sup> Depurar (escrutar y eliminar fallos).

## CAPÍTULO 2: PROPUESTA DE SOLUCIÓN.

### Introducción.

En el presente capítulo se hace referencia a los procesos del negocio que se desean automatizar, lo cual posibilita que se tenga un mayor conocimiento del mismo, en vista de lograr un sistema que cumpla con las necesidades reales del cliente. Se describen las principales características del sistema que fundamenta esta investigación y se enumeran las funcionalidades del software que se propone construir.

### 2.1. Descripción del Flujo Actual de los Procesos.

Las ECU son los artefactos fundamentales en el desarrollo de un sistema de gestión. En el proyecto CICPC las ECU, luego de ser detalladas y terminadas por los analistas del proyecto, pasan por la revisión y aprobación del equipo de Calidad Interna, una vez aprobadas allí, se envían a ser revisadas y aprobadas por Calidad UCI y cuando son liberadas por esta, se les muestra al cliente, que es el encargado de su aprobación final. Las ECU aprobadas por el cliente están listas para implementarse, cuando se termina con su implementación se comienza la escritura de un Manual de Usuario por cada ECU. Los manuales son elaborados por los escritores técnicos y al igual que las ECU, pasan por la revisión y probación de Calidad Interna, Calidad UCI y por último el cliente. Para un mejor entendimiento de este proceso, se puede consultar el diagrama correspondiente en el Anexo1.

Para controlar el progreso de estas actividades, cada jefe de equipo envía partes diarios al líder o al planificador del proyecto. Toda esta información referente al estado en que se encuentran las ECU y los Manuales de Usuario está disponible para todos los miembros del proyecto en un repositorio. Se utiliza la herramienta Microsoft Excel para llevar el control de los diferentes estados por los que pasan estos artefactos, en un documento llamado Priorización de Casos de Uso donde se tiene el listado de todos los Casos de Uso (CU) divididos por etapas y en otras columnas de la tabla, aparecen los estados en los que puede estar un CU. El procedimiento que se sigue en el proyecto para darle seguimiento a los CU es marcar con una "x" el estado por el que se encuentra, dentro de los diferentes equipos, siendo la escritura del Manual de Usuario otro de esos estados, pero como se genera otro artefacto, estos también pasan por procesos similares que la ECU. El líder del proyecto es el encargado de actualizar diariamente el documento Priorización de Casos de Uso, con la información que recibe de los distintos jefes de equipo que trabajan en la elaboración del software. Por su parte cada jefe de equipo controla en documentos separados el cumplimiento de las actividades propias de su equipo, que toman como punto de partida la ECU. La actualización de dichos documentos se realiza

diariamente, con el objetivo de ver el porcentaje de avance del proyecto, información de vital importancia para la gestión y seguimiento del mismo. Sin embargo, el procedimiento empleado no brinda la posibilidad de gestionar este proceso de forma automatizada y es necesario de una supervisión y actualización constante por parte del líder del proyecto o la persona encargada de esta actividad.

## 2.2. Propuesta de Sistema.

Durante el desarrollo de los procesos mencionados anteriormente se pueden encontrar varios flujos de actividades que pueden ser automatizados, para contrarrestar la pérdida de información que pudiera generarse si se realizan manualmente.

Como propuesta de solución que fundamenta esta investigación se plantea la creación del *Sistema de Control de Estado de las Especificaciones de Casos de Uso y Manuales de Usuario del SIIPOL*, el mismo será una aplicación web, administrada por el CMS Drupal y su propósito fundamental será automatizar el flujo actual de los procesos descritos en el epígrafe anterior.

El sistema será accedido por los usuarios a través de una Red de Área Local (LAN según sus siglas en inglés), estará basado en tecnología PHP, con un sistema gestor de base de datos MySQL y un servidor de aplicaciones web Apache 2.2.8.

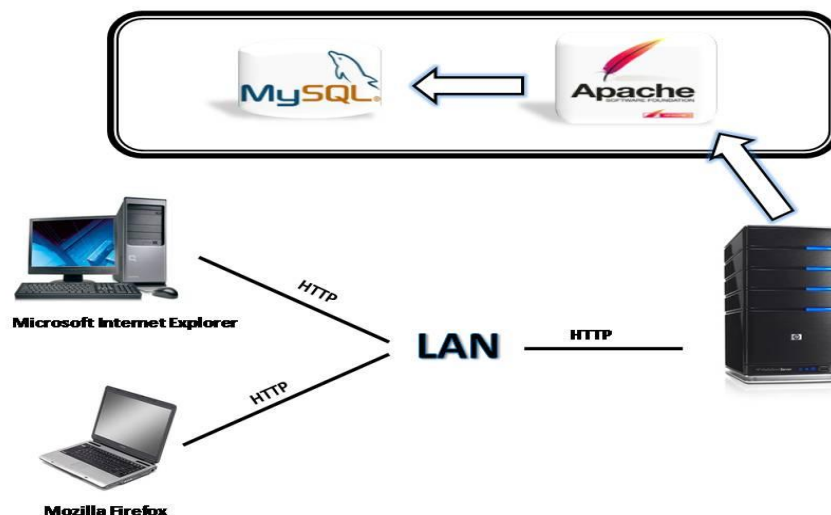
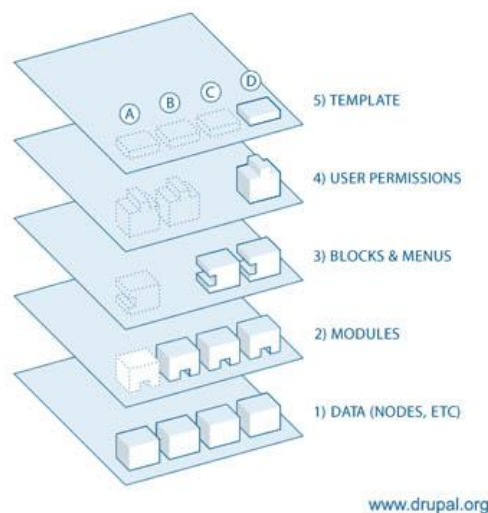


Figura 4. Propuesta del Sistema.

La arquitectura del sistema estará basada en el CMS con que se crea la aplicación. Drupal utiliza un estilo arquitectónico un tanto singular, que puede tener similitudes al Modelo-Vista-Controlador (MVC)



o al N-capas. Las similitudes vienen dadas porque Drupal hace una excelente separación limpia entre la persistencia de los datos y los formularios (modelo), la lógica (ciclo de respuesta al pedido http a un sitio Drupal) y la vista que emerge como la respuesta HTML al explorador de Internet (browser)[18]. Por otra parte, se puede decir que Drupal para conseguir su reconocida flexibilidad y facilidad en la creación de sitios web usa la abstracción y organización en capas que aplica en el tratamiento de los contenidos, estructurando los mismos en una serie de elementos básicos separados en diferentes capas, estos son los Nodos(Nodes), Módulos(Modules), Bloques y Menús(Blocks & Menus), Permisos de Usuario(User Permissions) y Plantillas(Templates), lo cual lo hace un sistema de gestión de contenidos modular y muy configurable[19].



**Figura 5. Capas de Drupal.**

### **2.3. Usuarios del Sistema.**

El sistema propuesto en esta investigación está dirigido a facilitar el trabajo de los jefes de equipo del proyecto CICPC, por lo que sólo será accedido por usuarios que tendrán un rol determinado, estos roles se corresponden con cada uno de jefes de equipo del proyecto, es decir, el líder del proyecto, el analista principal, el jefe de Calidad Interna, el de Calidad UCI, el representante del cliente, el jefe de los desarrolladores, el escritor técnico principal y el administrador del sistema, encargado de darle mantenimiento al mismo. Los roles de dichos usuarios y las funciones que realizarán los mismos se detallan a continuación:

Tabla 1. Funciones de los usuarios del sistema.

Roles del sistema.	Función que Realiza.
Analista	Es el encargado de la gestión (incluir, ver, modificar, eliminar) de ECU. No tendrá permisos limitados sobre esta entidad por lo que podrá cambiar su estado, consultar, filtrar listados de Casos de Uso que tenga asignados y hacer reportes de lo mostrado en pantalla.
Escritor Técnico	Es el encargado de la gestión de Manuales de Usuario. Por lo que también se le permitirá cambiar el estado de los mismos, consultar y filtrar listados de Manuales de Usuario que tenga asignados y hacer reportes de lo mostrado en pantalla.
Jefe de equipo	Dentro de esta categoría estarán los roles responsables de Calidad Interna, Calidad UCI, el Cliente y el Jefe del Equipo de Desarrollo. A este tipo de usuario se le permitirá cambiar los estados por los que se encuentra los Casos de Uso o Manuales de Usuarios. Podrá filtrar el listado de los mismos que tenga asignado en su módulo y hacer reportes de lo que se muestra en pantalla.
Líder de Proyecto	Se le permitirá hacer consultas y realizar reportes en el sistema de todos los Casos de Uso y Manuales de Usuario.
Administrador	Será el encargado de la gestión de los usuarios del sistema, así como el mantenimiento y configuración del mismo.

## 2.4. Requisitos del Software.

Los requisitos de software son condiciones o capacidades que necesita un usuario para resolver un problema o alcanzar un objetivo[20]. Estos se clasifican en funcionales y no funcionales, aportando la visión general de lo que se quiere hacer en el sistema.

### 2.4.1. Funcionalidades del sistema.

Los requisitos funcionales se definen como las condiciones o capacidades que el sistema debe cumplir[21]. Producto a que la metodología XP genera muy poca cantidad de artefactos, no se definen como tal los requisitos funcionales del sistema, sino un listado de las funcionalidades que serán implementadas. Sin embargo, teniendo en cuenta, que el documento de esta investigación servirá de bibliografía para una mejor comprensión del sistema a desarrollar, se ha decidido detallar cada una de

las funcionalidades con el objetivo de conseguir un mejor entendimiento de las mismas. Por lo tanto, considerando las metas trazadas con el sistema propuesto, este debe ser capaz de cumplir con lo siguiente:

### **F 1. Gestionar Cuenta de Usuario.**

- F 1.1. Incluir usuario.
- F 1.2. Modificar usuario.
- F 1.3. Eliminar usuario.
- F 1.4. Ver datos del usuario.
- F 1.5. Asignar roles al usuario.

### **F 2. Gestionar Roles.**

- F 2.1. Incluir rol.
- F 2.2. Modificar rol.
- F 2.3. Eliminar rol.
- F 2.4. Asignar permisos al rol.

### **F 3. Autenticar Usuario.**

- F 3.1. Mostrar el formulario de autenticación del usuario.
- F 3.2. Pedir al usuario introducir los datos para su autenticación: nombre, contraseña y pregunta de seguridad.
- F 3.3. Validar los datos introducidos.
  - F 3.3.a Mostrar mensaje en caso de error durante la validación.

### **F 4. Insertar Caso de Uso.**

- F 4.1. Mostrar el formulario para incluir el Caso de Uso.
- F 4.2. Permitir al usuario introducir los datos del Caso de Uso.
- F 4.3. Validar los datos introducidos por el usuario.
  - F 4.3.a Mostrar mensaje de error al usuario en caso de existir algún problema con la validación de los datos.

**F 5. Modificar Caso de Uso.**

F 5.1. Mostrar el listado de Casos de Uso existentes en el equipo de trabajo.

F 5.2. Mostrar el formulario para incluir el Caso de Uso.

F 5.3. Permitir al usuario modificar los datos del Caso de Uso que desee.

F 5.4. Validar los datos introducidos por el usuario.

F 5.4.a Mostrar mensaje de error al usuario en caso de existir algún problema con la validación de los datos.

**F 6. Ver datos del Caso de Uso.**

F 6.1. Permitir al usuario seleccionar el Caso de Uso del cual quiere ver sus datos.

F 6.2. Permitir al usuario modificar los datos del Caso de Uso que desee, en dependencia de sus permisos.

**F 7. Eliminar Caso de Uso.**

F 7.1. Mostrar al usuario un listado con todos los Casos de Uso existentes.

F 7.2 Permitir al usuario seleccionar el Caso de Uso que quiere eliminar.

F 7.3 Permitir al usuario eliminar el Caso de Uso.

F 7.3.a Mostrar al usuario un mensaje de éxito de la operación.

F 7.3.b Mostrar al usuario un mensaje de error en caso de que ocurra algún problema.

**F 8. Filtrar Caso de Uso.**

F 8.1. Mostrar al usuario un listado con todos los Casos de Uso existentes.

F 8.2. Mostrar al usuario un formulario con los datos para realizar la búsqueda.

F 8.3. Permitir al usuario introducir los datos de búsqueda.

F 8.4. Validar los datos introducidos por el usuario.

F 8.4.a Mostrar mensaje de error al usuario en caso de existir algún problema con la validación de los datos.

**F 9. Cambiar Estado de Caso de Uso.**

F 9.1. Mostrar al usuario un listado con todos los Casos de Uso existentes.

F 9.2 Permitir al usuario seleccionar el Caso de Uso que quiere cambiar el estado.

F 9.3. Mostrar al usuario un formulario con el campo del estado de forma editable.

F 9.4. Permitir al usuario cambiar el estado.

F 9.5. Validar los datos introducidos por el usuario.

F 9.5.a Mostrar mensaje de error al usuario en caso de existir algún problema con la validación de los datos.

#### **F 10. Asignar Caso de Uso.**

F 10.1. Mostrar al usuario un listado con todos los Casos de Uso existentes.

F 10.2 Permitir al usuario seleccionar el Caso de Uso que desea asignar.

F 10.3. Permitir al usuario asignar el Caso de Uso a un responsable.

F 10.4. Validar los datos introducidos por el usuario.

F 10.4.a. Mostrar mensaje de error al usuario en caso de existir algún problema con la validación de los datos.

#### **F 11. Insertar Manual de Usuario.**

F 11.1. Mostrar el formulario para incluir el Manual de Usuario.

F 11.2. Permitir al usuario introducir los datos del Manual de Usuario.

F 10.3. Validar los datos introducidos por el usuario.

F 11.3.a Mostrar mensaje de error al usuario en caso de existir algún problema con la validación de los datos.

#### **F 12. Modificar Manual de Usuario.**

F 12.1. Mostrar el formulario para incluir el Manual de Usuario.

F 12.2. Permitir al usuario modificar los datos del Manual de Usuario que desee.

F 12.3. Validar los datos introducidos por el usuario.

F 12.3.a Mostrar mensaje de error al usuario en caso de existir algún problema con la validación de los datos.

#### **F 13. Ver datos del Manual de Usuario.**

F 13.1. Permitir al usuario seleccionar el Manual de Usuario del cual quiere ver sus datos.

F 13.2. Permitir al usuario modificar los datos del Manual de Usuario que desee, si tiene permisos para hacerlo.

**F 14. Eliminar Manual de Usuario.**

F 14.1. Mostrar al usuario un listado con todos los manuales de usuario existentes.

F 14.2 Permitir al usuario seleccionar el Manual de Usuario que quiere eliminar.

F 14.3 Permitir al usuario eliminar el Manual de Usuario.

F 13.3.a Mostrar al usuario un mensaje de éxito de la operación.

F 14.3.b Mostrar al usuario un mensaje de error en caso de que ocurra algún problema.

**F 15. Filtrar Manual de Usuario.**

F 11.1. Mostrar al usuario un listado con todos los manuales de usuario existentes.

F 15.2. Mostrar al usuario un formulario con los datos para realizar la búsqueda.

F 14.3. Permitir al usuario introducir los datos de búsqueda.

F 15.4. Validar los datos introducidos por el usuario.

F 14.a Mostrar mensaje de error al usuario en caso de existir algún problema con la validación de los datos.

**F 16. Cambiar Estado de Manual de Usuario.**

F 16.1. Mostrar al usuario un listado con todos los manuales de usuario existentes.

F 16.2 Permitir al usuario seleccionar el Manual de Usuario que quiere cambiar el estado.

F 16.3. Mostrar al usuario un formulario con el campo del estado de forma editable.

F 16.4. Permitir al usuario cambiar el estado.

F 16.5. Validar los datos introducidos por el usuario.

F 16.5.a Mostrar mensaje de error al usuario en caso de existir algún problema con la validación de los datos.

**F 17. Asignar Manual de Usuario.**

F 17.1. Mostrar al usuario un listado con todos los manuales de usuario existentes.

F 17.2 Permitir al usuario seleccionar el Manual de Usuario que desea asignar.

F 17.3. Permitir al usuario asignar el Manual de Usuario a un responsable.

F 17.4. Validar los datos introducidos por el usuario.

F 17.4.a Mostrar mensaje de error al usuario en caso de existir algún problema con la validación de los datos.

**F 18. Generar Reporte.**

F 18.1. Mostrar formulario con los datos con los que se desee hacer el reporte.

F 18.2. Mostrar la opción que permita exportar los datos a PDF.

**F 19. Consultar Casos de Uso Asignados.**

F 19.1. Mostrar al usuario un formulario con todos los Casos de Uso que tiene asignados a su módulo.

F 19.2. Permitir al usuario ordenar el listado alfabéticamente.

F 19.3. Permitir al usuario asignar un Caso de Uso seleccionado.

F 19.4. Permitir al usuario cambiar el estado de un Caso de Uso seleccionado.

F 19.5. Permitir al usuario modificar un Caso de Uso seleccionado (sólo si tiene permiso para hacerlo).

**F 20. Consultar Manuales de Usuario Asignados.**

F 20.1. Mostrar al usuario un formulario con todos los manuales de usuario que tiene asignados a su módulo.

F 20.2. Permitir al usuario ordenar el listado alfabéticamente.

F 20.3. Permitir al usuario asignar un Manual de Usuario seleccionado.

F 20.4. Permitir al usuario cambiar el estado de un Manual de Usuario seleccionado.

F 20.5. Permitir al usuario modificar un Manual de Usuario seleccionado (sólo si tiene permiso para hacerlo).

### 2.4.2. Otras características del sistema.

Para dar soporte a las funcionalidades del sistema, se definieron además, características no funcionales (CNF) que deberá cumplir el mismo. Las cuales harán al producto, atractivo, usable, rápido o confiable.

#### **CNF 1. Restricciones en el diseño y la implementación.**

- Usar el lenguaje de programación PHP 5.2.6.
- Usar el gestor de Base de Datos MySQL 5.0.51.
- Utilizar como servidor web Apache 2.2.8.
- Desarrollar bajo el CMS Drupal 5.X.
- Utilizar como metodología de desarrollo XP.

#### **CNF 2. Interfaz.**

- No debe tener animaciones ni imágenes pesadas, lo cual repercute en la rapidez de la aplicación.
- El diseño de la interfaz de usuario debe llamar la atención del mismo sin llegar a ser agresivo a la vista.

#### **CNF 3. Usabilidad.**

- El sistema podrá ser usado por cualquier usuario que tenga conocimientos básicos de computación.

#### **CNF 4. Software.**

- Máquina con sistemas Operativos Windows 98 o superior o Linux.
- Usar un navegador web como: Mozilla Firefox (versión 3.5.0 o superior), Internet Explorer (versión 7.0 o superior).

#### **CNF 5. Hardware.**

- PC Pentium 2 o superior.
- Se requiere de un servidor con mínimo de 256 MB de RAM y 10 GB de espacio libre en disco duro.
- Tanto la PC servidor como las clientes deben de estar conectadas a la red.

#### **CNF 5. Rendimiento.**

- Debe ser rápida, eficiente tanto en los tiempos de respuesta como en la velocidad de procesamiento.



### **CNF 6. Confidencialidad.**

- La información manejada por el sistema debe ser protegida de accesos no autorizados.

### **CNF 7. Seguridad.**

- Asegurar que el usuario sea autenticado antes de realizar cualquier operación sobre la aplicación.
- Asignar roles a los usuarios permitiendo que sólo tengan acceso a las operaciones de acuerdo con los permisos que tienen.
- Sólo el administrador tiene el control total del sistema.
- Sólo debe tener acceso a la información el personal del proyecto autorizado.

### **CNF 8. Portabilidad.**

- El sistema debe ser multiplataforma.

## **Conclusiones.**

En este capítulo se hizo referencia a las principales características que tendrá el sistema a desarrollar. Se definieron las funcionalidades que cumplirá el software. Y teniendo definidas las mismas ya se puede comenzar con la construcción de la propuesta de solución.

## CAPÍTULO 3: EXPLORACIÓN Y PLANIFICACIÓN.

### Introducción.

En el presente capítulo se hace referencia a las fases de Exploración y Planificación del proyecto de software que presenta esta investigación, propias de la metodología XP, bajo la cual se desarrolla. Se detallan además los artefactos obtenidos en dichas etapas del desarrollo.

### 3.1. Fase de Exploración.

La fase de Exploración es la primera de las fases que plantea la metodología XP. En esta fase, los clientes plantean a grandes rasgos las Historias de Usuario que son de interés para la primera entrega del producto. Al mismo tiempo, el equipo de desarrollo se familiariza con las herramientas, tecnologías y prácticas que se utilizarán en el proyecto[22].

#### 3.1.1. Historias de Usuario.

Las Historias de Usuario son descripciones sencillas de las funcionalidades del software. Son escritas por el cliente empleando un lenguaje claro. Se emplearán para hacer estimaciones de tiempo y para realizar el plan de entregas[22]. A continuación se detallan las historias de usuario realizadas por el cliente durante la fase de Exploración del proyecto, que abarcan las funcionalidades del sistema propuesto.

Tabla 2. HU Gestionar Usuario.

Historia de Usuario	
<b>Número: 1</b>	<b>Usuario:</b> Administrador
<b>Nombre historia:</b> Gestionar Usuario.	
<b>Prioridad en negocio:</b> Baja.	<b>Riesgo en desarrollo:</b> Baja.
<b>Puntos estimados:</b> 1	<b>Iteración asignada:</b> 1
<b>Programador responsable:</b> Alexey Casanova Alvarez-Ivonne Guerra Anaya	
<b>Descripción:</b> Permite administrar las cuentas de los usuarios del sistema, posibilitando incluir, modificar y eliminar las mismas según se estime conveniente. A	

<p>cada usuario se le asignará un rol, al que se le darán los permisos que necesite en dependencia de su responsabilidad en el sistema.</p>
<p><b>Observaciones:</b></p>
<p><b>Referencias:</b> F 1</p>

Tabla 3. HU Gestionar Rol de Usuario.

Historia de Usuario	
<b>Número: 2</b>	<b>Usuario:</b> Administrador.
<b>Nombre historia:</b> Gestionar Rol de Usuario.	
<b>Prioridad en negocio:</b> Alta.	<b>Riesgo en desarrollo:</b> Baja.
<b>Puntos estimados:</b> 1	<b>Iteración asignada:</b> 1
<b>Programador responsable:</b> Alexey Casanova Alvarez-Ivonne Guerra Anaya	
<b>Descripción:</b> Permite gestionar los roles que tendrán los usuarios del sistema. A cada rol se le editarán los permisos para acceder a los distintos módulos. Debe permitir además insertar, modificar y eliminar roles.	
<b>Observaciones:</b>	
<b>Referencias:</b> F 2	

Tabla 4. HU Autenticar Usuario.

Historia de Usuario	
<b>Número: 3</b>	<b>Usuario:</b> Administrador.
<b>Nombre historia:</b> Autenticar Usuario.	
<b>Prioridad en negocio:</b> Alta.	<b>Riesgo en desarrollo:</b> Baja.

<b>Puntos estimados:</b> 1	<b>Iteración asignada:</b> 1
<b>Programador responsable:</b> Alexey Casanova Alvarez-Ivonne Guerra Anaya	
<b>Descripción:</b> Se debe permitir antes de iniciar una sesión por el usuario en la aplicación, que se solicite el nombre de usuario y su clave para que tenga acceso a los datos que corresponden a su rol de usuario. Se manejarán varios usuarios con distintos permisos de ingreso a los menús de acceso correspondientes, los cuales cuentan con privilegios distintos. Además, se responderá una pregunta de seguridad.	
<b>Observaciones:</b>	
<b>Referencias:</b> F 3	

Tabla 5. HU Insertar Caso de Uso.

Historia de Usuario	
<b>Número:</b> 4	<b>Usuario:</b> Analista.
<b>Nombre historia:</b> Insertar Caso de Uso.	
<b>Prioridad en negocio:</b> Alta.	<b>Riesgo en desarrollo:</b> Alta.
<b>Puntos estimados:</b> 1	<b>Iteración asignada:</b> 1
<b>Programador responsable:</b> Alexey Casanova Alvarez-Ivonne Guerra Anaya	
<b>Descripción:</b> Cuando se desea incluir una nueva ECU, el usuario debe autenticarse previamente y seleccionar la acción en el menú de insertar una nueva ECU, de la cual se solicitan datos como: Nombre, Módulo, Sub Módulo, Iteración, Estado y Responsable.	
<b>Observaciones:</b>	

Referencias: F 4

Tabla 6. HU Mostrar listado de Casos de Uso.

Historia de Usuario	
<b>Número: 5</b>	<b>Usuario:</b> Todos los usuarios.
<b>Nombre historia:</b> Mostrar listado de Casos de Uso.	
<b>Prioridad en negocio:</b> Media.	<b>Riesgo en desarrollo:</b> Media.
<b>Puntos estimados:</b> 1	<b>Iteración asignada:</b> 1
<b>Programador responsable:</b> Alexey Casanova Alvarez-Ivonne Guerra Anaya	
<b>Descripción:</b> Se le mostrará el usuario el listado de todos los CU asignados a su equipo de trabajo, permitiendo a partir del mismo, filtrar el listado por criterios y ver detalles de cada CU.	
<b>Observaciones:</b>	
Referencias: F 4	

Tabla 7. HU Modificar Caso de Uso.

Historia de Usuario	
<b>Número: 6</b>	<b>Usuario:</b> Analista.
<b>Nombre historia:</b> Modificar Caso de Uso.	
<b>Prioridad en negocio:</b> Alta.	<b>Riesgo en desarrollo:</b> Alta.
<b>Puntos estimados:</b> 2	<b>Iteración asignada:</b> 2
<b>Programador responsable:</b> Alexey Casanova Alvarez-Ivonne Guerra Anaya	

<p><b>Descripción:</b> Cuando se quiere modificar los datos de una ECU previamente incluida, se muestran los campos de los datos: Nombre, Módulo, Sub Módulo, Iteración, Responsable y Estado, de manera editable permitiendo realizar los cambios. El sistema debe permitir guardar los cambios realizados, así como el usuario que los hizo y la fecha en que se realizó la modificación.</p>
<p><b>Observaciones:</b> Se debe seleccionar previamente la ECU que se desea modificar.</p>
<p><b>Referencias:</b> F 5</p>

Tabla 8. HU Modificar Caso de Uso.

Historia de Usuario	
<b>Número:</b> 7	<b>Usuario:</b> Analista.
<b>Nombre historia:</b> Eliminar Caso de Uso.	
<b>Prioridad en negocio:</b> Media.	<b>Riesgo en desarrollo:</b> Alta.
<b>Puntos estimados:</b> 1	<b>Iteración asignada:</b> 2
<b>Programador responsable:</b> Alexey Casanova Alvarez-Ivonne Guerra Anaya	
<p><b>Descripción:</b> Cuando el usuario desea eliminar una ECU, debe autenticarse previamente y seleccionar en el menú la opción de consultar el listado de ECU, luego seleccionar la ECU que se desea eliminar y seleccionar la opción que le permitirá eliminarla del sistema.</p>	
<p><b>Observaciones:</b> Se debe seleccionar previamente la ECU que se desea eliminar.</p>	
<p><b>Referencias:</b> F 7</p>	

Tabla 9. HU Filtrar Listado de Casos de Uso.

Historia de Usuario
---------------------

<b>Número: 8</b>	<b>Usuario:</b> Todos los usuarios.
<b>Nombre historia:</b> Filtrar Listado de Casos de Uso.	
<b>Prioridad en negocio:</b> Media.	<b>Riesgo en desarrollo:</b> Media.
<b>Puntos estimados:</b> 2	<b>Iteración asignada:</b> 3
<b>Programador responsable:</b> Alexey Casanova Alvarez-Ivonne Guerra Anaya	
<b>Descripción:</b> Se debe mostrar el listado de todos los CU asignados al equipo de trabajo y se debe permitir introducir los datos: Nombre, Módulo, Sub Módulo, Iteración y Estado para realizar la búsqueda, una vez completada esta acción, se mostrarán los CU coincidentes con los criterios de búsqueda seleccionados.	
<b>Observaciones:</b> Se debe introducir al menos un criterio de búsqueda	
<b>Referencias:</b> F 8	

Tabla 10. HU Cambiar Estado de Caso de Uso.

Historia de Usuario	
<b>Número: 9</b>	<b>Usuario:</b> Jefe de Equipo.
<b>Nombre historia:</b> Cambiar Estado de Caso de Uso.	
<b>Prioridad en negocio:</b> Alta.	<b>Riesgo en desarrollo:</b> Media.
<b>Puntos estimados:</b> 1	<b>Iteración asignada:</b> 2
<b>Programador responsable:</b> Alexey Casanova Alvarez-Ivonne Guerra Anaya	
<b>Descripción:</b> Se mostrará el listado de todos los CU asignados al módulo y se debe permitir cambiar los valores del campo Estado, del CU que se seleccione. Una vez que se seleccione cambiar el estado se debe asignar el Caso de Uso a alguna persona.	

<b>Observaciones:</b> Se debe seleccionar el previamente el Caso de Uso.
<b>Referencias:</b> F 9 y F 10

Tabla 11. HU Consultar CU.

Historia de Usuario	
<b>Número: 10</b>	<b>Usuario:</b> Analista.
<b>Nombre historia:</b> Consultar CU.	
<b>Prioridad en negocio:</b> Alta.	<b>Riesgo en desarrollo:</b> Media.
<b>Puntos estimados:</b> 1	<b>Iteración asignada:</b> 3
<b>Programador responsable:</b> Alexey Casanova Alvarez-Ivonne Guerra Anaya	
<b>Descripción:</b> Se mostrará el listado de todos los Casos de Uso aunque no estén asignados a su módulo. Se debe permitir además modificar su estado, ver datos del Caso de Uso, ordenar el listado alfabéticamente, modificar el CU (si tiene los permisos para hacerlo)	
<b>Observaciones:</b> El Analista es el único que puede modificar el CU	
<b>Referencias:</b> F 5, F 6, F 8 y F 19	

Tabla 12. HU Incluir Manual de Usuario.

Historia de Usuario	
<b>Número: 11</b>	<b>Usuario:</b> Escritor Técnico.
<b>Nombre historia:</b> Incluir Manual de Usuario.	
<b>Prioridad en negocio:</b> Alta.	<b>Riesgo en desarrollo:</b> Alta.
<b>Puntos estimados:</b> 1	<b>Iteración asignada:</b> 1



<b>Programador responsable:</b> Alexey Casanova Alvarez-Ivonne Guerra Anaya
<b>Descripción:</b> Cuando se desea incluir un nuevo Manual de Usuario, el usuario debe autenticarse previamente y seleccionar la acción en el menú de registrar un nuevo Manual de Usuario, en la cual especificará datos como: Nombre, Módulo, Sub Módulo, Iteración, Responsable y Estado.
<b>Observaciones:</b>
<b>Referencias:</b> F 11

Tabla 13. HU Mostrar listado de Manuales de Usuario.

Historia de Usuario	
<b>Número: 12</b>	<b>Usuario:</b> Todos los usuarios.
<b>Nombre historia:</b> Mostrar listado de Manuales de Usuario.	
<b>Prioridad en negocio:</b> Media.	<b>Riesgo en desarrollo:</b> Media.
<b>Puntos estimados:</b> 1	<b>Iteración asignada:</b> 1
<b>Programador responsable:</b> Alexey Casanova Alvarez-Ivonne Guerra Anaya	
<b>Descripción:</b> Se le mostrará el usuario el listado de todos los Manuales de Usuario asignados a su equipo de trabajo, permitiendo a partir del mismo, filtrar el listado por criterios y ver detalles de cada Manual.	
<b>Observaciones:</b>	
<b>Referencias:</b> F 11	

Tabla 14. HU Modificar Manual de Usuario.

Historia de Usuario	
<b>Número: 13</b>	<b>Usuario:</b> Escritor Técnico.

<b>Nombre historia:</b> Modificar Manual de Usuario.	
<b>Prioridad en negocio:</b> Alta.	<b>Riesgo en desarrollo:</b> Alta.
<b>Puntos estimados:</b> 1	<b>Iteración asignada:</b> 2
<b>Programador responsable:</b> Alexey Casanova Alvarez-Ivonne Guerra Anaya	
<b>Descripción:</b> Cuando se quiere modificar los datos de un Manual de Usuario. Se muestran los campos de datos: Nombre, Módulo, Sub Módulo, Iteración, Responsable y Estado, de manera editable permitiendo realizar los cambios. El sistema debe permitir guardar los cambios realizados, así como el usuario que los hizo y la fecha en que se realizó la modificación.	
<b>Observaciones:</b> Se debe seleccionar previamente el Manual de Usuario que se desea modificar.	
<b>Referencias:</b> F 12	

Tabla 15. HU Eliminar Manual de Usuario.

Historia de Usuario	
<b>Número:</b> 14	<b>Usuario:</b> Escritor Técnico.
<b>Nombre historia:</b> Eliminar Manual de Usuario.	
<b>Prioridad en negocio:</b> Alta.	<b>Riesgo en desarrollo:</b> Alta.
<b>Puntos estimados:</b> 1	<b>Iteración asignada:</b> 2
<b>Programador responsable:</b> Alexey Casanova Alvarez-Ivonne Guerra Anaya	
<b>Descripción:</b> Cuando el Escritor Técnico desea eliminar un Manual de Usuario, debe autenticarse previamente y seleccionar en el menú la opción de consultar el listado de manuales de usuario, luego seleccionar el manual de usuario que se desea eliminar y seleccionar la opción que le permitirá eliminarlo del sistema.	

<b>Observaciones:</b>
<b>Referencias:</b> F 14

Tabla 16. HU Filtrar Listado de Manuales de Usuario.

Historia de Usuario	
<b>Número:</b> 15	<b>Usuario:</b> Jefe de Equipo.
<b>Nombre historia:</b> Filtrar Listado de Manuales de Usuario.	
<b>Prioridad en negocio:</b> Alta.	<b>Riesgo en desarrollo:</b> Media.
<b>Puntos estimados:</b> 2	<b>Iteración asignada:</b> 3
<b>Programador responsable:</b> Alexey Casanova Alvarez-Ivonne Guerra Anaya	
<b>Descripción:</b> Se debe mostrar el listado de todos los manuales de usuario asignados y se debe permitir introducir los datos: Nombre, Módulo, Sub Módulo, Iteración, Responsable y Estado para realizar la búsqueda. Se mostrarán los manuales coincidentes con los criterios de búsqueda seleccionados.	
<b>Observaciones:</b> Se debe introducir al menos un criterio de búsqueda	
<b>Referencias:</b> F 13 y F 15	

Tabla 17. HU Cambiar Estado de Manual de Usuario.

Historia de Usuario	
<b>Número:</b> 16	<b>Usuario:</b> Jefe de Equipo.
<b>Nombre historia:</b> Cambiar Estado de Manual de Usuario.	
<b>Prioridad en negocio:</b> Alta	<b>Riesgo en desarrollo:</b> Baja

<b>Puntos estimados:</b> 1	<b>Iteración asignada:</b> 2
<b>Programador responsable:</b> Alexey Casanova Alvarez-Ivonne Guerra Anaya	
<b>Descripción:</b> Se debe mostrar el listado de todos los manuales asignados y se debe permitir cambiar los valores del campo Estado, del Manual de Usuario que se seleccione. Una vez que se seleccione cambiar el estado se debe asignar el Manual de Usuario a alguna persona.	
<b>Observaciones:</b>	
<b>Referencias:</b> F 16 y F 17	

Tabla 18. HU Consultar Manuales.

Historia de Usuario	
<b>Número:</b> 17	<b>Usuario:</b> Escritor Técnico.
<b>Nombre historia:</b> Consultar Manuales.	
<b>Prioridad en negocio:</b> Alta.	<b>Riesgo en desarrollo:</b> Media.
<b>Puntos estimados:</b> 2	<b>Iteración asignada:</b> 3
<b>Programador responsable:</b> Alexey Casanova Alvarez-Ivonne Guerra Anaya	
<b>Descripción:</b> Se mostrará el listado de todos los Manuales de Usuario, aunque no estén asignados a su módulo. Se debe permitir además modificar su estado, ver todos sus datos, ordenar el listado alfabéticamente, modificar el Manual de Usuario.	
<b>Observaciones:</b> El Escritor Técnico es el único que puede modificar el Manual de Usuario.	

**Referencias:** F 13, F 15 y F 19

**Tabla 19. HU Generar Reporte.**

Historia de Usuario	
<b>Número:</b> 18	<b>Usuario:</b> Jefe de Equipo.
<b>Nombre historia:</b> Generar Reporte.	
<b>Prioridad en negocio:</b> Media.	<b>Riesgo en desarrollo:</b> Alta.
<b>Puntos estimados:</b> 2	<b>Iteración asignada:</b> 3
<b>Programador responsable:</b> Alexey Casanova Alvarez-Ivonne Guerra Anaya	
<b>Descripción:</b> Se debe permitir exportar a formato PDF el listado de Casos de Uso o Manuales de Usuario mostrados en pantalla.	
<b>Observaciones:</b>	
<b>Referencias:</b> F 18	

### 3.2. Fase de Planificación.

La metodología XP plantea que la planificación es un diálogo continuo entre las partes involucradas en el proyecto. Durante la fase de Planificación se priorizan las historias de usuario y se acuerda el alcance de la entrega[22]. Los programadores estiman cuánto esfuerzo requiere cada historia y a partir de allí se define el cronograma. Las estimaciones del esfuerzo asociado a la implementación de las historias la establecen los programadores, utilizando como medida el punto. Un punto, equivale a una semana ideal de programación[23] y a partir de allí se determina la velocidad del proyecto, la cual es utilizada para establecer cuántas historias se pueden implementar antes de una fecha determinada o cuánto tiempo tomará implementar un conjunto de historias.

### 3.2.1 Estimación del esfuerzo por Historias de Usuario.

En conjunto con el cliente y todas las partes implicadas en la construcción del software propuesto se realizó una estimación para cada una de las Historias de Usuario, posibilitando tener una visión global del tiempo de desarrollo del sistema. Los resultados obtenidos se muestran a continuación:

**Tabla 20. Puntos de Estimación por Historia de Usuario**

Historias de Usuario	Puntos de Estimación
Gestionar Usuario.	1
Gestionar Rol de Usuario.	1
Autenticar Usuario.	1
Insertar Caso de Uso.	1
Mostrar listado de Casos de Uso.	1
Modificar Caso de Uso.	2
Eliminar Caso de Uso.	1
Filtrar Listado de Casos de Uso.	1
Cambiar Estado de Caso de Uso.	1
Consultar CU Asignados.	1
Incluir Manual de Usuario.	1
Mostrar listado de Manuales de usuario.	1
Modificar Manual de Usuario.	1
Eliminar Manual de Usuario.	1
Filtrar Listado de Manuales de Usuario.	1
Cambiar Estado de Manual de Usuario.	1
Consultar Manuales Asignados.	1
Generar Reporte.	2

### 3.2.2. Plan de Iteraciones.

Dentro de la planificación del proyecto se seleccionan las historias de usuario que se implementarán primero de acuerdo con su prioridad y cuál será el tiempo que se empleará en cada etapa. Esta información está recogida dentro del Plan de Iteraciones, el cual especifica las historias de usuario seleccionadas para cada entrega de acuerdo con el orden preestablecido. Para el desarrollo del sistema propuesto se han definido 3 iteraciones, las cuales se describen a continuación:

#### **Iteración 1**

La primera iteración del desarrollo del sistema, tendrá como objetivo la implementación de las funcionalidades básicas y significativas para la puesta en marcha del mismo. Se tendrán en cuenta las historias de usuario 1, 2, 3, 4, 5, 11 y 12. De manera general al finalizar esta iteración se habrá podido cumplir con funcionalidades como la gestión de usuarios con sus respectivos roles y permisos, la autenticación de los mismos y se podrá incluir Casos de Uso y Manuales de Usuarios en el sistema, obteniéndose la primera versión del software que se le entregará al cliente.

#### **Iteración 2**

Durante la segunda iteración se seguirán implementando historias de usuario de prioridad alta como la 6, 7, 9, 12, 14 y 16. Al finalizar esta iteración, quedarán automatizados en el sistema los procesos de modificar y eliminar los datos de los Casos de Uso y Manuales de Usuario y se podrá además, cambiar el estado por los que pasan los mismos. 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.

#### **Iteración 3**

Durante la tercera iteración se desarrollarán las restantes historias de usuario, la 8, 10, 15, 17 y 18. Con las cuales se completarán todas las funcionalidades para las que está diseñado el sistema. Se podrán consultar, filtrar y ver los datos de los Casos de Uso, Manuales de Usuario y permitirá generar reportes en PDF de la información mostrada en pantalla. Finalizando esta iteración el sistema estará listo para entregarse al cliente completamente. Se le incluirán las modificaciones de interfaz que necesite y luego esta versión del software será desplegada, para poner a prueba su funcionamiento bajo condiciones reales.

**3.2.3. Plan de Duración de las Iteraciones.**

Para conocer el tiempo real de la duración de las iteraciones del proyecto, así como las historias de usuario que serán implementadas en cada una de las mismas, la metodología XP propone la creación del Plan de Duración de las Iteraciones. Su propósito fundamental es conocer el tiempo que abarcará cada iteración y el orden en que serán implementadas las historias de usuario dentro de cada período.

**Tabla 21. Plan de Duración de las Iteraciones.**

Iteraciones	Historias de usuario que se implementan	Duración
<b>Iteración 1</b>	<ol style="list-style-type: none"> <li>1. Gestionar Usuario.</li> <li>2. Gestionar Rol de Usuario.</li> <li>3. Autenticar Usuario.</li> <li>4. Insertar Caso de Uso.</li> <li>5. Mostrar listado de Casos de Uso.</li> <li>6. Incluir Manual de Usuario.</li> <li>7. Mostrar listado de Manuales de Usuario.</li> </ol>	7 semanas
<b>Iteración 2</b>	<ol style="list-style-type: none"> <li>1. Modificar Caso de Uso.</li> <li>2. Eliminar Caso de Uso.</li> <li>3. Cambiar Estado de Caso de Uso.</li> <li>4. Modificar Manual de Usuario.</li> <li>5. Eliminar Manual de Usuario.</li> <li>6. Cambiar Estado de Manual de Usuario.</li> </ol>	6 semanas
<b>Iteración 3</b>	<ol style="list-style-type: none"> <li>1. Filtrar Listado de Casos de Uso.</li> <li>2. Consultar CU Asignados.</li> <li>3. Filtrar Listado de Manuales de Usuario.</li> <li>4. Consultar Manuales Asignados.</li> <li>5. Generar Reporte.</li> </ol>	6 semanas



### 3.2.4. Plan de Entrega.

El Plan de Entrega no es más que el cronograma de las entregas de partes funcionales del software que serán hechas al cliente. Las pequeñas entregas del sistema serán hechas en las siguientes fechas:

Tabla 22. Plan de Entrega de las funcionalidades por Iteración.

Iteraciones.	Funcionalidades que Abarca.	Fecha de Entrega.
<b>1ra Iteración</b>	<ul style="list-style-type: none"> <li>• Gestión y autenticación de Usuarios.</li> <li>• Inclusión de ECU y Manuales de Usuario al sistema.</li> </ul>	15 de enero del 2010
<b>2da Iteración</b>	<ul style="list-style-type: none"> <li>• Modificación y eliminación datos de ECU y Manuales de Usuario.</li> <li>• Cambiar estado de ECU y Manuales de Usuario.</li> </ul>	30 de marzo del 2010
<b>3ra Iteración</b>	<ul style="list-style-type: none"> <li>• Consultar listado de ECU y Manuales de Usuarios asignados.</li> <li>• Filtrar los resultados de las consultas.</li> <li>• Generar Reportes.</li> </ul>	3 de mayo del 2010

### Conclusiones.

En este capítulo se ha hecho referencia a las fases de Exploración y Planificación del sistema que fundamenta la presente investigación. Se han descrito los artefactos más importantes que la metodología de desarrollo propone para esta etapa y que sirven de base para las próximas etapas de desarrollo. Las fases que se detallaron en este capítulo son de gran importancia, pues marcan las pautas de la construcción del software, lográndose estimar el tiempo que tomará implementarlo, así como la prioridad de las versiones del sistema que serán entregadas al cliente, organizando de forma general el trabajo del equipo de desarrollo, por lo que se puede decir que ya están todas las condiciones creadas para comenzar con la implementación de la aplicación.

## CAPÍTULO 4: DISEÑO E IMPLEMENTACIÓN.

### Introducción

En el presente capítulo se hace referencia al diseño de la aplicación y de la base de datos como punto de partida para la implementación del sistema. Se describen las tareas realizadas durante las tres iteraciones del desarrollo del software que se definieron anteriormente, de forma tal, que al finalizar cada una de ellas se entregará una parte funcional del software al cliente y este realizará las pruebas pertinentes para su aceptación final.

### 5.1. Diseño del Sistema.

Para hablar del diseño del sistema propuesto, se debe comenzar haciendo una breve descripción del diseño del CMS Drupal, sobre el cual será publicada la aplicación. Drupal está dividido en varios paquetes: los nodes, que son los elementos básicos en que el CMS almacena la información o contenidos. Los modules, que son los elementos que operan sobre los nodes y otorgan funcionalidad a Drupal permitiendo incrementar sus capacidades o adaptarlas a las necesidades de cada sitio web. Por último los themes, que tienen la función de establecer la apariencia gráfica o estilo de la información que se le muestra al usuario. Esta separación entre información y aspecto gráfico, permite cambiar el diseño o apariencia del sitio web sin necesidad de modificar los contenidos, lo que es muy práctico si lo único que se necesita es renovar la apariencia de un sitio web.

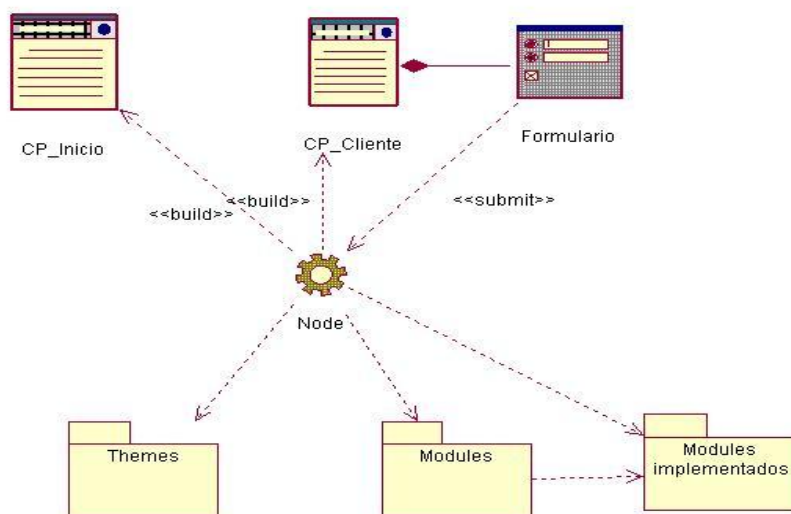
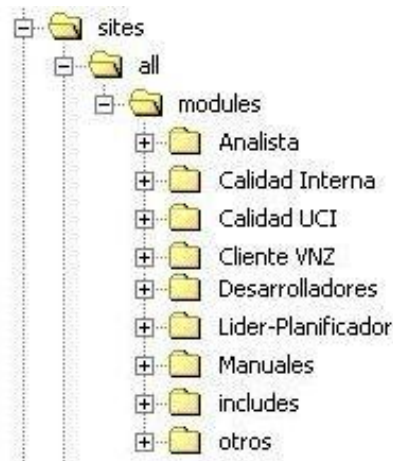


Figura 6. Diagrama de clases de diseño de Drupal.

Para el desarrollo de la aplicación propuesta se implementaron otros módulos que se integraron a los que trae instalados por defecto Drupal, con el objetivo de implementar en ellos todas las funcionalidades que aportará el sistema, detalladas previamente en capítulos anteriores de este trabajo.



**Figura 7. Nuevos módulos implementados.**

La metodología XP propone que todo proyecto que se realice tenga un diseño lo más sencillo posible. Para esto define el uso de tarjetas CRC (Class, Responsibilities and Collaboration, por sus siglas en inglés), las que permiten al programador apreciar el desarrollo orientado a entidades específicas. A partir de estas premisas, se diseñaron un total de siete tarjetas CRC, una por cada nuevo módulo implementado, donde se presenta el módulo, las funcionalidades que abarca y el resto de los módulos relacionados a este. Las tarjetas CRC se detallaron a partir de cada módulo producto a que las funcionalidades en Drupal se implementan dentro de cada módulo y no dentro de una clase previamente definida por el programador. Todas las tarjetas CRC especificadas, se muestran a continuación, con una breve explicación de las funciones de cada uno de los módulos que abarcan.

El módulo Analista será el encargado de la gestión total de todas las funcionalidades que se pueden realizar con los Casos de Uso. En él, se comienza el flujo de los procesos a partir de la inclusión en el sistema de un nuevo CU, se podrán realizar todas las actividades para las que está diseñado el sistema tomando en cuenta esta entidad, es decir, será el único módulo encargado de la modificación de la entidad Caso de Uso, así como su eliminación total del sistema. También se podrá acceder a otras funcionalidades comunes a otros módulos como son, exportar a PDF los listados mostrados en pantalla y cambiar el estado en que se encuentra la entidad. La tarjeta CRC correspondiente se muestra a continuación:

Tabla 23. Tarjeta CRC módulo Analista.

Tarjeta CRC	
Módulo Analista	
Funcionalidades	Colaboraciones(Módulos)
Insertar Caso de Uso.	Node
Modificar Caso de Uso.	Menú
Eliminar Caso de Uso.	User
Filtrar Listado de Casos de Uso.	Calidad Interna
Cambiar Estado de Caso de Uso.	Calidad UCI
Consultar CU.	Cliente VNZ
Generar Reportes.	Desarrolladores
Mostrar Listado de CU.	Manual

Las tarjetas CRC que se muestran a continuación corresponden a los módulos de Calidad Interna, Calidad UCI, Desarrolladores y Cliente VNZ. Con el acceso a estos módulos sólo se podrá obtener permisos para cambiar el estado de las entidades, filtrar y generar reportes con los listados mostrados en pantalla. Los estados aportados por cada uno de estos módulos son distintos, los cuales indicarán la evolución y progreso de los Casos de Uso y Manuales de Usuario, según sea el caso.

Tabla 24. Tarjeta CRC módulo Calidad Interna.

Tarjeta CRC	
Módulo Calidad Interna	
Funcionalidades	Colaboraciones(Módulos)
Filtrar Listado de Casos de Uso.	Node
Cambiar Estado de Caso de Uso.	Menú
Generar Reportes.	User
Mostrar Listado de CU.	Analista
Cambiar Estado de Manual de usuario.	Manual
Mostrar Listado de Manual de Usuario.	

Tabla 25. Tarjeta CRC módulo Calidad UCI.

Tarjeta CRC	
Módulo Calidad UCI	
Funcionalidades	Colaboraciones(Módulos)
Filtrar Listado de Casos de Uso.	Node
Cambiar Estado de Caso de Uso.	Menú
Generar Reportes.	User
Mostrar Listado de CU.	Analista

Tabla 26. Tarjeta CRC módulo Desarrolladores.

Tarjeta CRC	
Módulo Desarrolladores	
Funcionalidades	Colaboraciones(Módulos)
Filtrar Listado de Casos de Uso.	Node
Cambiar Estado de Caso de Uso.	Menú
Generar Reportes.	User
Mostrar Listado de CU.	Analista

Tabla 27. Tarjeta CRC módulo Cliente VNZ.

Tarjeta CRC	
Módulo Cliente VNZ	
Funcionalidades	Colaboraciones(Módulos)
Filtrar Listado de Casos de Uso.	Node
Cambiar Estado de Caso de Uso.	Menú
Generar Reportes.	User
Mostrar Listado de CU.	Analista

El módulo Manual va a ser el encargado y responsable de la gestión completa de los Manuales de Usuario, desde su inclusión a su eliminación en el sistema. También se tendrá acceso a cambiar sus estados, exportar a PDF los listados mostrados, consultar y filtrar los mismos de acuerdo con varios criterios de búsqueda. Dentro de este módulo comienza el ciclo por el que pasan los manuales en el sistema. La tarjeta CRC correspondiente se muestra a continuación:

Tabla 28. Tarjeta CRC módulo Manual.

Tarjeta CRC	
Módulo Manual	
Funcionalidades	Colaboraciones(Módulos)
Incluir Manual de Usuario.	Node
Modificar Manual de Usuario.	Menú
Eliminar Manual de Usuario.	User
Filtrar Listado de Manuales de Usuario.	Calidad Interna
Cambiar Estado de Manual de Usuario.	
Consultar Manuales.	
Generar Reportes.	
Mostrar listado de Manuales de Usuario.	

Finalmente, dentro del módulo Líder –Planificador sólo se podrá consultar los CU y Manuales de Usuario existentes en el sistema, así como filtrar y exportar a PDF los listados mostrados. Por lo que no se tendrá acceso a modificar o eliminar ninguna de las entidades. Este módulo será accedido por el líder o el planificador del proyecto, para conocer el progreso de cada CU o Manual de Usuario a partir del porcentaje de avance y estados que tengan los mismos.

**Tabla 29. Tarjeta CRC módulo Líder-Planificador.**

<b>Tarjeta CRC</b>	
<b>Módulo Líder-Planificador</b>	
<b>Funcionalidades</b>	<b>Colaboraciones(Módulos)</b>
Filtrar Listado de Casos de Uso. Consultar CU. Generar Reportes. Consultar Manuales.	Node Menú User Analista
Filtrar Listado de Manuales de Usuario.	
Mostrar listado de Manuales de Usuario.	

**4.1.1 Diseño de la Base de Datos.**

Uno de los pasos cruciales en la construcción de una aplicación que maneje datos, es sin duda, el diseño de la base de datos, de manera que se tenga un acceso fácil y rápido a la información, con redundancia mínima. Para lograr este propósito, se definieron las clases persistentes que utilizará el sistema para lograr un correcto funcionamiento del mismo. El diagrama de clases correspondiente se muestra a continuación:

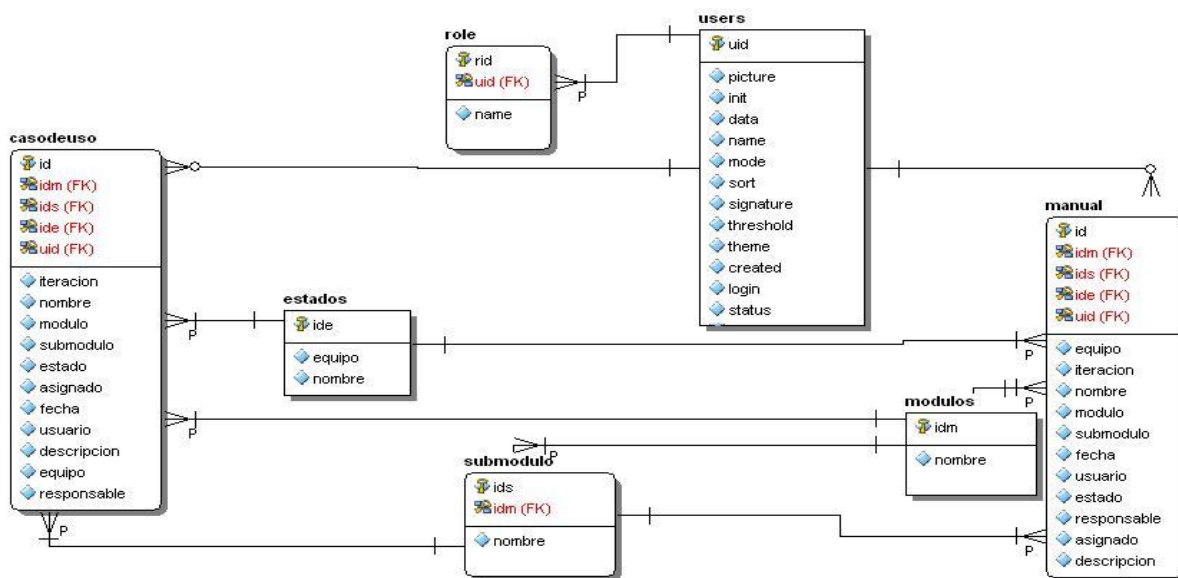


Figura 8. Diagrama Entidad-Relación de la BD.

## 4.2. Fase de Implementación.

Como se ha detallado anteriormente un proyecto desarrollado en XP se divide en iteraciones para realizar la codificación, de manera que se puede entregar una versión funcional del software al cliente, que aunque no esté completo en su totalidad, irá creciendo ciclo a ciclo. En cada iteración, se implementarán las historias de usuario definidas previamente en el Plan de Iteraciones. Se define además un conjunto de tareas, que deberán realizar los programadores para llevar a cabo la construcción del sistema.

### Implementación de la 1ra Iteración.

Durante la primera iteración de la construcción del software se implementaron las historias de usuario básicas para poner en marcha el sistema, donde se obtuvo la primera versión del mismo, que posteriormente fue aprobada por el cliente. En esta iteración se le dio cumplimiento a varias tareas detalladas en el Anexo 2.

### Implementación de la 2da Iteración.

Durante la implementación de la segunda iteración del software se perfeccionaron las funcionalidades de la primera entrega y se continuó con el desarrollo de las historias de usuario de prioridad alta, organizando el proceso de acuerdo con las tareas especificadas en el Anexo 3.

### Implementación de la 3ra Iteración.

Una vez concluida la implementación de las historias de usuario de la tercera iteración de la construcción del software, se obtuvo la versión completa del producto para ser entregado al cliente y listo para usarse en condiciones reales. Las tareas que se realizaron para el logro de este objetivo se pueden observar en el Anexo 4.

#### 4.2.1. Diagrama de Despliegue.

El diagrama de despliegue no es más que la distribución física que va a tener la aplicación una vez desplegada a los usuarios. El diagrama correspondiente a la distribución de la aplicación será el siguiente:

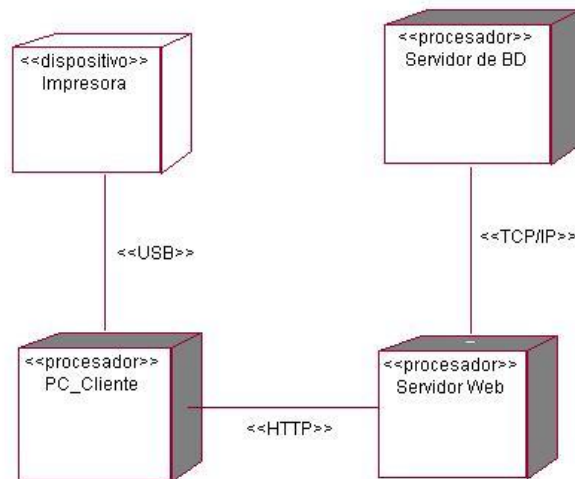


Figura 9. Diagrama de Despliegue.

### Conclusiones.

En este capítulo se definió el diseño del sistema para ser implementado y se construyó el diagrama de clases persistentes que utilizará la base de datos. Las tareas definidas posibilitaron implementar la aplicación de una forma más organizada. Al concluir esta etapa se obtuvo la versión completa del “Sistema de Control de Estado de las Especificaciones de Casos de Uso y Manuales de Usuario del SIIPOL”, con lo que se da por terminado el desarrollo de la solución propuesta.



## CAPÍTULO 5: VALIDACIÓN DE LA PROPUESTA.

### Introducción

No se puede asegurar que algo funcione a menos que se pruebe, las pruebas unitarias aseguran que el desarrollador elimine esta incertidumbre. Las pruebas de aceptación aseguran que el cliente tiene el producto que necesita de acuerdo con los requerimientos planteados[24]. En este capítulo se hace referencia a las pruebas efectuadas a través de las cuales se validará que el producto cuenta con la calidad requerida y satisface las expectativas de sus usuarios finales.

#### 4.4.1. Pruebas Realizadas.

Para validar el cumplimiento de las funcionalidades del software construido, se realizaron pruebas de aceptación. Como la metodología XP propone, estas pruebas fueron realizadas por el cliente y son básicamente, pruebas funcionales sobre el sistema, que se detallan a partir de las historias de usuario, definiéndose un conjunto de casos de prueba. Es recomendable definir un caso de prueba por cada historia, con el objetivo de garantizar que todas las funcionalidades queden cubiertas, aunque se pueden diseñar tantos casos de prueba por historia como se desee. Los casos de prueba más importantes, realizados para probar el funcionamiento de la aplicación se detallan en el Anexo 5. El código también fue probado por los programadores cada vez que se añadían nuevas funcionalidades, utilizando un plugin del NetBeans que permitió debuggear el nuevo código, integrado al resto de la aplicación.

#### 4.4.2. Resultados de las Pruebas.

Como se ha descrito anteriormente, el sistema fue realizado en 3 iteraciones, al final de las cuales se realizaban las respectivas pruebas de aceptación por parte del cliente, con el objetivo de efectuar las entregas de las funcionalidades previamente pactadas y que estas se correspondieran con lo descrito en el listado de requisitos anteriormente definido y detallado con mayor profundidad en las historias de usuario.

Al concluir el proceso de pruebas de cada iteración, se documentó la cantidad de *No Conformidades (NC)* encontradas. Las NC se clasificaron de acuerdo a su tipo y a su complejidad como se muestra en la siguiente tabla:

Tabla 30. Clasificación de las NC

Clasificación de NC	
Tipo	Complejidad
1- Error de Funcionalidad	1- Alta
2- Error de Interfaz	2- Media
3- Ortografía	3- Baja
4- Redacción	
5- Mensajes del Sistema	
6- Recomendaciones	

Para medir los resultados obtenidos durante las pruebas se hizo un recuento de todas las NC que arrojaron las diferentes iteraciones, donde se observó un decremento de la cantidad de errores encontrados, en la medida en que se fueron completando las funcionalidades del sistema. El número de NC por cada iteración se muestra en la siguiente figura:

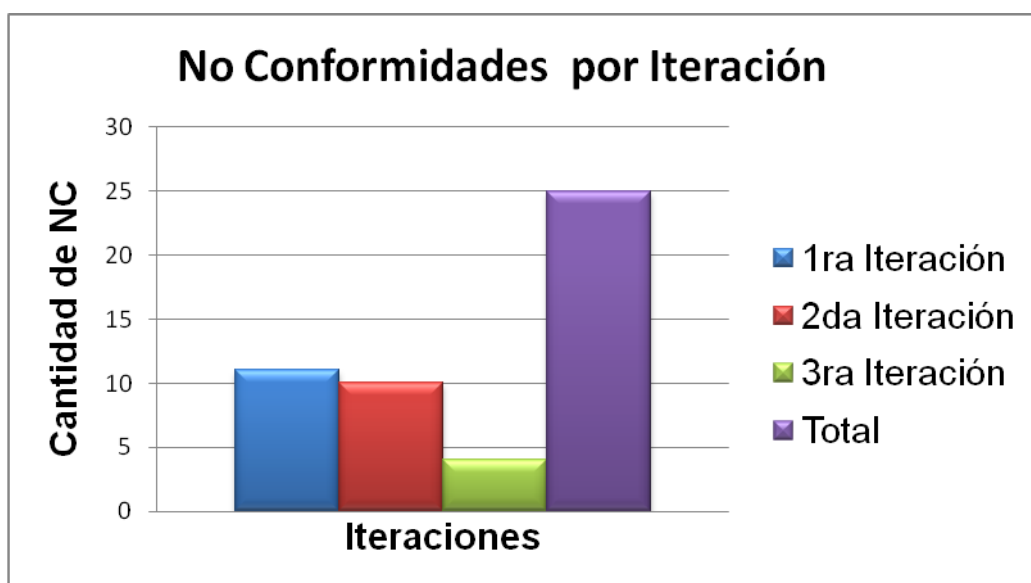


Figura 10. Cantidad de no conformidades por iteración.

Al clasificar los defectos encontrados por tipo, se evidenció que la mayoría de las inconformidades del cliente eran sobre la forma de mostrar la información en pantalla y la navegación entre las interfaces. También aportó varias recomendaciones que fueron tomadas en cuenta y modificadas en el sistema con el interés de lograr una mayor conformidad de sus usuarios.

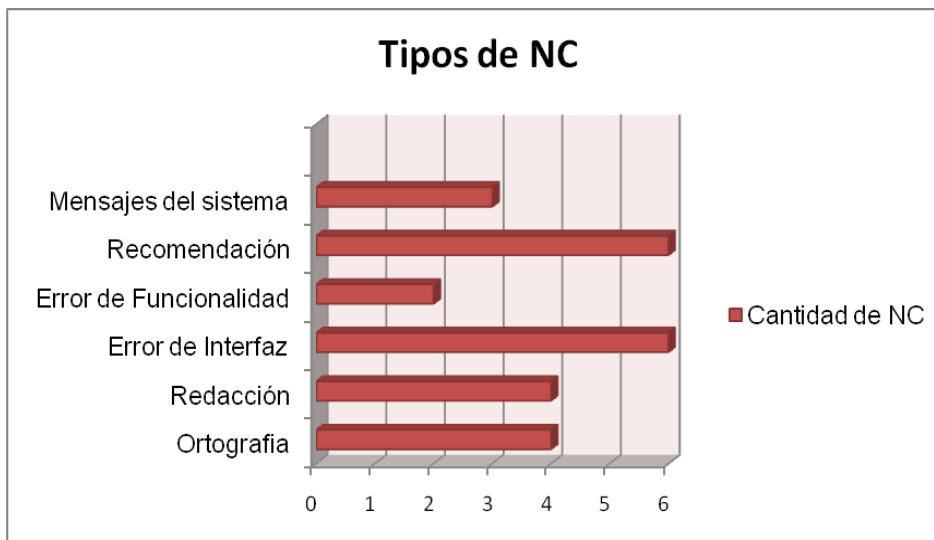


Figura 11. Clasificación de las NC por tipo.

#### 4.4.3. Valoración del producto.

La herramienta implementada tuvo muy buena aceptación por parte del cliente, todas las NC fueron resueltas, presentando en cada iteración de pruebas, un sistema más sólido y funcional. El Acta de Aceptación emitida por el cliente, una vez aceptada la versión completa de la aplicación construida se muestra en el Anexo 6.

Es importante destacar, que el sistema, una vez implementado y aplicándolo al trabajo del líder del proyecto CICPC apoyará en gran medida, el control que este realiza dentro del proceso de desarrollo, permitiendo tener una visión global del estado del mismo. Su uso evitaría posibles errores humanos y la dependencia de una tercera persona, cuyo trabajo sea asignar las tareas del siguiente rol. Con esta herramienta se agilizará el flujo de información entre los diferentes roles, factor que ha atentado en otras etapas de desarrollo contra el proyecto, provocando atrasos en el mismo.

Para obtener una mayor visión de las ventajas de la aplicación del producto construido, se decidió realizar una encuesta a los usuarios que en el futuro van a interactuar con el sistema. Los usuarios seleccionados para la aplicación de la encuesta fueron los jefes de los diferentes equipos de trabajo del proyecto CICPC, incluyendo al líder y al sub-líder, los cuales son personas que poseen varios años de experiencia de trabajo dentro del mismo. Se mostró a dichos usuarios la aplicación, explicándoles todas las funcionalidades que brindaba y se les pidió que estos evaluaran en un rango del 1 al 5 el grado de adecuación de la herramienta al trabajo que desempeñan. Se les explicó que la puntuación de 5 corresponde a la evaluación de Excelente, 4 a la de Muy Bien, 3 a la de Bien, 2 a la de Regular y

por último el 1, correspondía a la categoría de Mal. El cuestionario realizado puede observarse en el Anexo 7.

Luego de recopilar todas las respuestas se hizo un resumen de los principales planteamientos y recomendaciones obtenidas, lo cual permitió la retroalimentación del equipo de desarrollo del nuevo sistema. Se obtuvo además la puntuación dada por los usuarios a cada pregunta, la cual se puede observar en la siguiente figura:

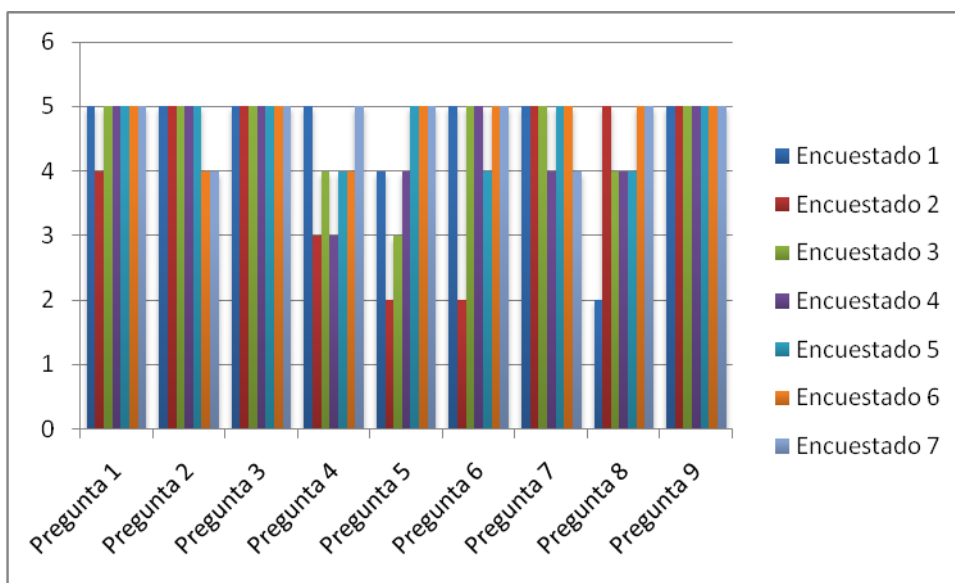


Figura 12. Puntuación por pregunta dada por los encuestados.

Una vez contabilizado el total de los puntos recibidos en cada pregunta, se definió una escala para evaluar el grado de adecuación de la herramienta al trabajo que realizará con ella sus futuros usuarios, la cual se puede observar en la siguiente tabla:

Tabla 31. Escala de evaluación de las preguntas del cuestionario.

Escala de Evaluación	
Rango de puntos	Adecuación
Menos de 25	Poco adecuado
25-30	Adecuado
30-35	Muy adecuado

El total de puntos que obtuvieron los criterios evaluados en cada pregunta puede observarse en la siguiente figura:

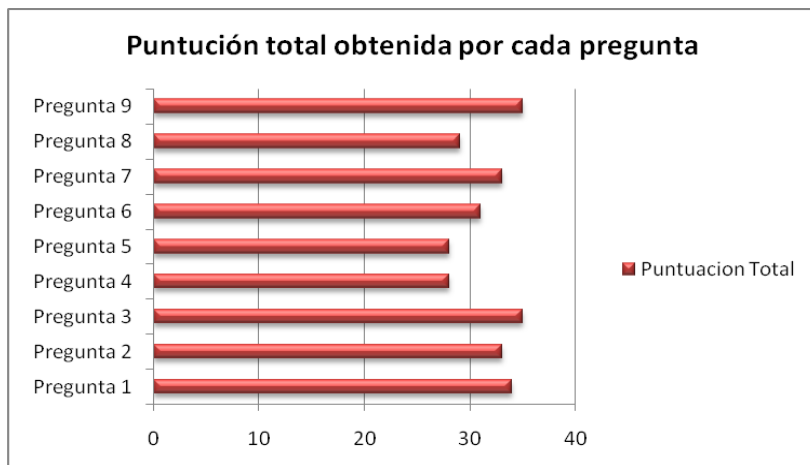


Figura 13. Total de puntos por cada criterio evaluado en cada pregunta.

Como se puede apreciar en el gráfico mostrado y tomando en consideración la escala de evaluación propuesta, todos los criterios fueron evaluados entre adecuados y muy adecuados, por lo que se puede concluir que el nuevo sistema tuvo éxito ante sus usuarios.

Las principales opiniones recogidas sobre la herramienta fueron las siguientes:

1. La herramienta desarrollada no es compleja de usar y brinda información acertada y oportuna.
2. Facilita el mecanismo de comunicación de las actividades que se pueden realizar con los Casos de Uso y Manuales de Usuario
3. Facilita el control del estado por el que se encuentra un Caso de Uso como entidad básica en el ciclo de desarrollo del proyecto.
4. Todos los usuarios consideran recomendable el uso de la herramienta en el proyecto.

## Conclusiones.

En este capítulo se abordó el tema de las pruebas de aceptación realizadas, como parte de la validación del sistema propuesto. Se definieron los casos de prueba de aceptación que sirvieron para guiar al cliente en cuanto a las funcionalidades del sistema para ser aceptado y que el mismo tuviera la mayor conformidad con lo desarrollado. Todo el proceso de pruebas sirvió para obtener un producto con la calidad requerida, el cual se encuentra listo para ser utilizado por el líder del proyecto CICPC y el resto de los jefes de los diferentes equipos de trabajo.

## CONCLUSIONES GENERALES

Los resultados obtenidos al concluir esta investigación, aportan la creación de una nueva herramienta para facilitar el trabajo realizado por el líder del proyecto CICPC, pues le permite mantener un control del proyecto, de manera que pueda obtener información inmediata, la que es utilizada para generar los informes solicitados por la dirección del centro de producción y darle seguimiento a las actividades realizadas en el proyecto.

La aplicación fue desarrollada siguiendo los principios de la metodología XP, lo que permitió realizar un desarrollo ágil y dentro de los plazos previstos con la total conformidad del cliente. Los artefactos obtenidos, sirven de base para una futura ampliación del sistema, en vista de continuar incrementando sus funcionalidades para seguir adaptándolo a las necesidades reales del entorno donde se utilizará.

Todo el trabajo realizado posibilitó darle cumplimiento al objetivo general de la investigación con la creación del *“Sistema de Control de Estado de las Especificaciones de Casos de Uso y Manuales de Usuario del SIIPOL”*, aportando a través de la misma mayor eficiencia en el control del trabajo que realizan todos los jefes de equipo del proyecto, posibilitando de esta forma mejorar los procesos organizativos de la información referente a los Casos de Uso y Manuales de Usuario del proyecto CICPC.

## **RECOMENDACIONES**

Después de todo el estudio realizado para la construcción de la aplicación propuesta y en vista de seguir perfeccionando la herramienta en el futuro, para lograr una mayor eficiencia en el trabajo que se realiza con ella, sería recomendable tener en cuenta los siguientes aspectos.

1. Ampliar la aplicación añadiéndole nuevas funcionalidades, como la gestión y planificación del tiempo y el manejo de otras entidades, con el objetivo de superar las expectativas de los usuarios que la utilizan.
2. Proponer la utilización de la herramienta en otros proyectos productivos de la Universidad de las Ciencias Informáticas con características similares al CICPC.
3. Migrar para el gestor de base de datos PostgreSQL, para continuar con la línea de desarrollo del software libre.

## REFERENCIAS BIBLIOGRÁFICAS:

1. Zalazar, I.C.N.J.R., *GS Proyect.Sistema Informático para la Gestión de Proyectos*, in *Facultad de Ingeniería Industrial.Centro de Estudio de Ingeniería y Sistemas Ingeniería Informática*. 2005, INSTITUTO SUPERIOR POLITÉCNICO “JOSÉ ANTONIO ECHEVERRÍA”: Ciudad de la Habana.
2. [cited 2009 diciembre]; Available from: <http://trac.edgewall.org/>.
3. [cited 2010 abril]; Available from: <http://www.unicordoba.edu.co/enlinea/solicitudcorreo/egroupware.html>.
4. *Clic Escuela 2.0*. [Cited 2010 enero]; Available from: <http://www.gobiernodecanarias.org/medusa/contenidosclicescuela2.0/index.asp@r0=tutoriales&r3=1.html>.
5. Netcraft. *Encuesta Servidores Web más utilizados*. 2009 [cited 2010 enero].
6. *Conferencias de Introducción a la Programación*. [cited 2010 enero]; Available from: <http://eva.uci.cu>.
7. *Curso Gestión y Tecnología*. 2009 [cited 2010 enero]; Available from: <http://www.mailxmail.com/curso-gestion-tecnologia/que-es-php>.
8. *Manual Sistema de Gestión de Base de Datos*. [cited 2009 noviembre]; Available from: <http://www.taringa.net/posts/ebooks-tutoriales/2641833/Manual-:-Sistema-de-Gesti%C3%B3n-de-Base-de-Datos-%5BEspa%C3%B1ol-PDF%5D.html>.
9. *Proceso Unificado de Rational* .[cited 2009 diciembre].
10. Beck, K. *Extreme Programming Explained* [cited 2010 Enero].
11. Jose H. Canós, P.L., María del Carmen Penadés. *Metodologías Ágiles en el Desarrollo de Software*. [cited 2010 marzo].
12. Amaro Calderón, S.D., J.C. Valverde Rebaza, and Trujillo. *Metodologías Ágiles*. 2007 [cited 2010 Enero].



13. Berrocal, J.L.A. *La Gestión de Contenidos Digitales: Acceso a la Información en el Ámbito Ciudadano*. [cited 2010 Enero]; Available from: <http://www.pliegosdeyuste.eu/n9pliegos/pdfs/115.pdf>.
14. Yeleny Serrano Barrero, E.P.T., *Proceso de desarrollo de aplicaciones web sobre la plataforma Drupal*. 2007, Universidad de las Ciencias Informáticas: Ciudad Habana.
15. *Definición de CSS*. [cited 2010 abril]; Available from: <http://www.masadelante.com/faqs/css>.
16. Evelio Ángel Aladro García, E.A.C., *Gestor Web de Información de Profesores de la Facultad 8*. 2009, Universidad de las Ciencias Informáticas: Ciudad de la Habana.
17. *Un sitio en Internet. Qué es la web?* [cited 2010 abril]; Available from: <http://www.monografias.com/trabajos5/laweb/laweb.shtml>.
18. Kane, V. *awebfactory*. 2008 [cited 2010 abril]; Available from: <http://lists.drupal.org/pipermail/development/2008-April/029590.html>.
19. *Arquitectura de Drupal*. [cited 2010 abril]; Available from: <http://www.comunic-art.com/portal/index.php/tutorial-drupal/arquitectura-de-drupal.html>.
20. *IEEE Standard Glossary of Software Engineering Terminology*. [cited 2010 marzo].
21. *Conferencias de Ingeniería de Software 1*. [cited 2010 marzo]; Available from: <http://eva.uci.cu>.
22. Villegas, A.A. *A Propósito de la programación extrema XP(eXtreme Programming)*. [cited 2010 marzo]; Available from: <http://www.monografias.com/trabajos51/programacion-extrema/programacion-extrema.shtml>.
23. [cited 2010 marzo]; Available from: <http://oness.sourceforge.net/proyecto/html/ch05.html#N102B1>.
24. Armas, M. *Extreme Programming*. [cited 2010 abril]; Available from: [http://www.liderdeproyecto.com/articulos/extreme\\_programming.html](http://www.liderdeproyecto.com/articulos/extreme_programming.html).

---

**BIBLIOGRAFÍA:**

1. **Barbero, Miguel Jaque.** Ventajas de la Programación Extrema en el entorno empresarial. [En línea]
2. **Beck, Kent.** *Extreme Programming Explained.*
3. **Berrocal, José Luis Alonso.** La Gestión de Contenidos Digitales: Acceso a la Información en el Ámbito Ciudadano. [En línea] <http://www.pliegosdeyuste.eu/n9pliegos/pdfs/115.pdf>.
4. **Calderón, Amaro.** 2007. *Metodologías Ágiles.* Perú : s.n., 2007.
5. **Emilio A. Sánchez, Patricio Letelier, José H. Conos.** *Mejorando la gestión de Historias de Usuario en eXtreme programming.* Valencia, España : s.n.
6. **Escribano, Gerardo Fernández.** 2002. *Introducción a Extreme Programing.* 2002.
7. **John K. VanDyk, Matt Westgate.** *Pro Drupal Development.*
8. **José H. Canós, Patricio Letelier, María del Carmen Penadés.** *Metodologías Ágiles en el Desarrollo de Software.* Valencia : s.n.
9. **Joskowicz, José.** 2008. *Reglas y Prácticas en Extreme Programming.* España : s.n., 2008.
10. *MANUAL DE USUARIO EGROUWARE.* s.l. : Universidad Tecnológica de Puebla.
11. **Molpeceres, Alberto.** 15-12-2002. *Procesos de desarrollo: RUP, XP, FDD.* 15-12-2002.
12. **Roberkys Martin Cruañez, Derick Enríquez Rey, Hubert Viltres Salas.** *Trabajo Investigativo sobre Sistemas de Gestión de Contenidos.* s.l. : Universidad de las Ciencias Informáticas.
13. **Sierra, Alejandro Aguilar.** *Introducción a la programación extrema.* s.l. : Universidad Autónoma de México.
14. **Solís, Manuel Calero.** *Una explicación de la programación extrema (XP).* Madrid : s.n.
15. **Villegas, Adrian Anaya.** A propósito de la programación extrema XP(eXtreme Programming). [En línea] <http://www.monografias.com/trabajos51/programacion-extrema/programacion-extrema.shtml>.
16. **Yeleny Serrano Barrero, Ekaterina Perdigón Torres.** *Proceso de desarrollo de Aplicaciones Web.* s.l. : Universidad de las Ciencias Informáticas.

## GLOSARIO DE TÉRMINOS

**API:** Conjunto de convenciones internacionales que definen cómo debe invocarse una determinada función de un programa desde una aplicación. Es el método que utiliza un programa para interactuar con funciones del sistema operativo o de la herramienta en la que se programa.

**CICPC:** Cuerpo de Investigaciones Científicas Penales y Criminalísticas.

**CMS:** Sistema de Gestión de Contenidos.

**Cookies:** Es un pedazo de información enviado por un servidor web a un buscador web, del cual se espera que el software del buscador web lo archive y lo envíe de regreso al servidor cada vez que el buscador requiere información adicional al servidor.

**CPU:** Unidad Central de Procesamiento. Es el cerebro de la computadora, ubicado específicamente en el microprocesador, posee dos componentes, la Unidad de Control y la Unidad aritmético-lógica. La CPU trabaja activamente con la memoria principal, aunque físicamente está separada del microprocesador.

**CU:** Caso de Uso. Recoge las funcionalidades a implementar en un software.

**Pluggin:** Programa que puede anexarse a otro para aumentar sus funcionalidades (generalmente sin afectar otras funciones ni afectar la aplicación principal). No se trata de un parche ni de una actualización, es un módulo aparte que se incluye opcionalmente en una aplicación.

**RAM:** Memoria de Acceso Aleatorio. Tipo de memoria donde la computadora guarda información para que pueda ser procesada más rápidamente. En la memoria RAM se almacena toda información que está siendo usada en el momento.

**SIIPOL:** Sistema Integrado de Información Policial. Aplicación que se desarrolla en el proyecto CICPC.

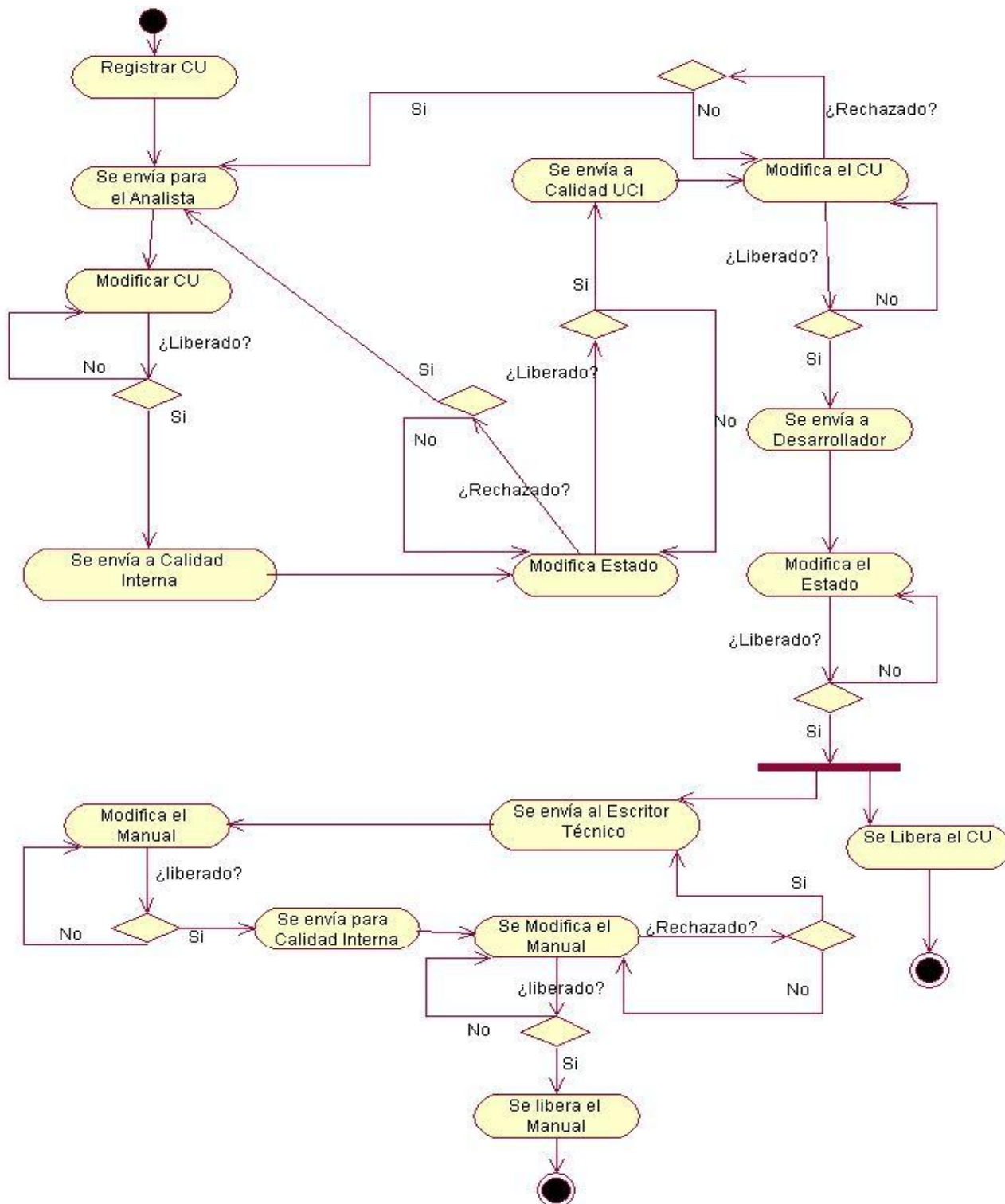
**Triggers:** Se define así a una subrutina que es ejecutada de manera automática cuando se produce algún tipo de transacción (inserción, borrado o actualización) en la tabla de una base de datos.

**Milestone:** Son metas o hitos. Pueden ser subproyectos creados en una herramienta de gestión de proyecto para organizar la información contenida en el mismo.

**Scripts:** Pequeños programas incrustados en las páginas que nos permiten definir interactividades de cualquier tipo.

**ANEXOS.**

**Anexo 1: Diagrama del proceso a automatizar.**



## Anexo 2: Tareas de Implementación de la primera iteración.

Tarea de Implementación	
<b>Número de tarea:</b> 1	<b>Número de HU:</b> 1 y 2
<b>Nombre de la tarea:</b> Configuración de los usuarios y sus permisos en sistema.	
<b>Tipo de tarea:</b> Configuración.	<b>Puntos estimados:</b> 1
<b>Fecha Inicio:</b> 1-2-2010	<b>Fecha Fin:</b> 3-2-2010
<b>Programador Responsable:</b> Alexey Casanova Alvarez-Ivonne Guerra Anaya	
<b>Descripción:</b> Se especificarán los usuarios del sistema, así como los roles y permisos que tendrán los mismos y que serán almacenados en la base de datos.	

Tarea de Implementación	
<b>Número de tarea:</b> 2	<b>Número de HU:</b> 3
<b>Nombre de la tarea:</b> Configuración de la autenticación de usuarios.	
<b>Tipo de tarea:</b> Configuración.	<b>Puntos estimados:</b> 1
<b>Fecha Inicio:</b> 3-2-2010	<b>Fecha Fin:</b> 6-2-2010
<b>Programador Responsable:</b> Alexey Casanova Álvarez-Ivonne Guerra Anaya	
<b>Descripción:</b> Se configurará el modo de autenticación de los usuarios del sistema, validando los datos introducidos con los almacenados en la base de datos.	

Tarea de Implementación	
<b>Número de tarea:</b> 3	<b>Número de HU:</b> 4 y 5
<b>Nombre de la tarea:</b> Modificación del código de la aplicación.	
<b>Tipo de tarea:</b> Desarrollo	<b>Puntos estimados:</b> 1
<b>Fecha Inicio:</b> 15-02-2010	<b>Fecha Fin:</b> 20-02-2010
<b>Programador Responsable:</b> Alexey Casanova Alvarez-Ivonne Guerra Anaya	
<b>Descripción:</b> Se comenzará con el desarrollo de las primeras funcionalidades que serán añadidas al sistema, que permitirá al usuario insertar Casos de Uso al sistema. Los datos de los mismos serán guardados en la base de datos. También se deberá mostrar un listado de los CU insertados.	

Tarea de Implementación	
<b>Número de tarea:</b> 4	<b>Número de HU:</b> 10 y 12
<b>Nombre de la tarea:</b> Modificación del código de la aplicación.	
<b>Tipo de tarea:</b> Desarrollo	<b>Puntos estimados:</b> 1
<b>Fecha Inicio:</b> 22-02-2010	<b>Fecha Fin:</b> 26-02-2010
<b>Programador Responsable:</b> Alexey Casanova Alvarez-Ivonne Guerra Anaya	
<b>Descripción:</b> Se continuará con el desarrollo de las primeras funcionalidades que serán añadidas al sistema, que permitirá al usuario insertar manuales de usuario. Los datos de los mismos serán guardados en la base de datos. Se mostrará también el listado de manuales insertados.	

<b>Tarea de Implementación</b>	
<b>Número de tarea:</b> 5	<b>Número de HU:</b> 4 y 11
<b>Nombre de la tarea:</b> Creación de tablas en la BD.	
<b>Tipo de tarea:</b> Desarrollo	<b>Puntos estimados:</b> 1
<b>Fecha Inicio:</b> 15-02-2010	<b>Fecha Fin:</b> 16-02-2010
<b>Programador Responsable:</b> Alexey Casanova Alvarez-Ivonne Guerra Anaya	
<b>Descripción:</b> Se gestionarán las nuevas tablas en la base de datos. Correspondientes a las entidades Casos de Uso y Manuales de Usuario.	

<b>Tarea de Implementación</b>	
<b>Número de tarea:</b> 6	<b>Número de HU:</b> 4 y 11
<b>Nombre de la tarea:</b> Definir formato de presentación de datos.	
<b>Tipo de tarea:</b> Desarrollo	<b>Puntos estimados:</b> 1
<b>Fecha Inicio:</b> 17-22-2010	<b>Fecha Fin:</b> 22-10-2010
<b>Programador Responsable:</b> Alexey Casanova Alvarez-Ivonne Guerra Anaya	
<b>Descripción:</b> Se definirá la forma de mostrar los datos de las entidades que se incluyan en pantalla.	

## Anexo 3: Tareas de Implementación de la segunda iteración.

Tarea de Implementación	
<b>Número de tarea:</b> 7	<b>Número de HU:</b> 6 ,7, 13 y 15
<b>Nombre de la tarea:</b> Modificación del código añadiendo nuevas funcionalidades.	
<b>Tipo de tarea:</b> Desarrollo	<b>Puntos estimados:</b> 3
<b>Fecha Inicio:</b> 1-03-2010	<b>Fecha Fin:</b> 18-03-2010
<b>Programador Responsable:</b> Alexey Casanova Alvarez-Ivonne Guerra Anaya	
<b>Descripción:</b> Se continuará perfeccionado las funcionalidades previamente implementadas y se le añadirán nuevas funcionalidades, como la modificación, eliminación de Casos de Uso y Manuales de Usuario del sistema.	

Tarea de Implementación	
<b>Número de tarea:</b> 8	<b>Número de HU:</b> 6 ,7, 13 y 15
<b>Nombre de la tarea:</b> Integración del código fuente.	
<b>Tipo de tarea:</b> Desarrollo	<b>Puntos estimados:</b> 1
<b>Fecha Inicio:</b> 18-03-2010	<b>Fecha Fin:</b> 10-03-2010
<b>Programador Responsable:</b> Alexey Casanova Alvarez-Ivonne Guerra Anaya	
<b>Descripción:</b> Se integrará el código generado de las nuevas funcionalidades al que estaba previamente implementado.	



Tarea de Implementación	
<b>Número de tarea:</b> 9	<b>Número de HU:</b> 6 ,7, 13 y 15
<b>Nombre de la tarea:</b> Configuración de los permisos de usuario.	
<b>Tipo de tarea:</b> Configuración	<b>Puntos estimados:</b> 1
<b>Fecha Inicio:</b> 19-03-2010	<b>Fecha Fin:</b> 19-02-2010
<b>Programador Responsable:</b> Alexey Casanova Alvarez-Ivonne Guerra Anaya	
<b>Descripción:</b> Se configurará los permisos de los usuarios de manera que puedan tener acceso a las nuevas funcionalidades implementadas.	

Tarea de Implementación	
<b>Número de tarea:</b> 10	<b>Número de HU:</b> 6 ,7, 13 y 15
<b>Nombre de la tarea:</b> Modificación de la interfaz de usuario	
<b>Tipo de tarea:</b> Desarrollo.	<b>Puntos estimados:</b> 1
<b>Fecha Inicio:</b> 10-03-2010	<b>Fecha Fin:</b> 18-03-2010
<b>Programador Responsable:</b> Alexey Casanova Alvarez-Ivonne Guerra Anaya	
<b>Descripción:</b> Se modificará la interfaz visual de la aplicación de manera que permita el acceso a las nuevas funcionalidades.	

## Anexo 4. Tareas de Implementación de la tercera iteración.

Tarea de Implementación	
<b>Número de tarea:</b>	<b>Número de HU:</b> 8,10, 16 y 17
<b>Nombre de la tarea:</b> Modificación del código fuente.	
<b>Tipo de tarea:</b> Desarrollo.	<b>Puntos estimados:</b> 3
<b>Fecha Inicio:</b> 22-03-2010	<b>Fecha Fin:</b> 5-04-2010
<b>Programador Responsable:</b> Alexey Casanova Alvarez-Ivonne Guerra Anaya	
<b>Descripción:</b> Se modificará el código añadiendo las restantes funcionalidades de consulta y filtrado de Casos de Uso y Manuales de Usuario.	

Tarea de Implementación	
<b>Número de tarea:</b> 11	<b>Número de HU:</b> 8,10, 16 y17
<b>Nombre de la tarea:</b> Modificación de la interfaz de usuario	
<b>Tipo de tarea:</b> Desarrollo.	<b>Puntos estimados:</b> 1
<b>Fecha Inicio:</b> 29-03-2010	<b>Fecha Fin:</b> 5-04-2010
<b>Programador Responsable:</b> Alexey Casanova Alvarez-Ivonne Guerra Anaya	
<b>Descripción:</b> Se modificará la interfaz visual de la aplicación de manera que permita el acceso a las nuevas funcionalidades.	

Tarea de Implementación	
<b>Número de tarea:</b>	<b>Número de HU:</b> 8, 10, 16 y 17
<b>Nombre de la tarea:</b> Configuración de los permisos de usuario.	
<b>Tipo de tarea:</b> Configuración	<b>Puntos estimados:</b> 1
<b>Fecha Inicio:</b> 6-04-2010	<b>Fecha Fin:</b> 6-04-2010
<b>Programador Responsable:</b> Alexey Casanova Alvarez-Ivonne Guerra Anaya	
<b>Descripción:</b> Se configurará los permisos de los usuarios de manera que puedan tener acceso a las nuevas funcionalidades implementadas.	

Tarea de Implementación	
<b>Número de tarea:</b> 12	<b>Número de HU:</b> 8, 10, 16 y 17
<b>Nombre de la tarea:</b> Integración del código fuente.	
<b>Tipo de tarea:</b> Desarrollo	<b>Puntos estimados:</b> 1
<b>Fecha Inicio:</b> 5-04-2010	<b>Fecha Fin:</b> 5-03-2010
<b>Programador Responsable:</b> Alexey Casanova Alvarez-Ivonne Guerra Anaya	
<b>Descripción:</b> Se integrará el código generado de las nuevas funcionalidades al que estaba previamente implementado.	

<b>Tarea de Implementación</b>	
<b>Número de tarea:</b> 13	<b>Número de HU:</b> 18
<b>Nombre de la tarea:</b> Generación del reporte.	
<b>Tipo de tarea:</b> Desarrollo	<b>Puntos estimados:</b> 2
<b>Fecha Inicio:</b> 19-4-2010	<b>Fecha Fin:</b> 26-4-2010
<b>Programador Responsable:</b> Alexey Casanova Alvarez-Ivonne Guerra Anaya	
<b>Descripción:</b> Se implementará y configurará la manera de generar los reportes del sistema en formato PDF.	

#### Anexo 5. Casos de Prueba de Aceptación.

<b>Caso de Prueba de Aceptación</b>	
<b>Código:</b> CP4_1	<b>Historia de Usuario:</b> 4
<b>Descripción:</b> Insertar un CU en el sistema.	
<b>Condiciones de Ejecución:</b> El usuario debe estar autenticado y tener permiso para realizar esta acción en el sistema.	
<b>Entrada/Pasos de Ejecución:</b> Se accede a la opción de insertar los datos de un nuevo CU en el sistema. Si se introducen datos inválidos o se deja algún campo vacío se debe mostrar un mensaje de error.	
<b>Resultado Esperado:</b> Se insertan los datos el CU en el sistema.	
<b>Evaluación de la Prueba:</b> Prueba satisfactoria.	

Caso de Prueba de Aceptación	
<b>Código:</b> CP5_1	<b>Historia de Usuario:</b> 15
<b>Descripción:</b> Mostrar listado de Casos de Uso.	
<b>Condiciones de Ejecución:</b> El usuario debe estar autenticado	
<b>Entrada/Pasos de Ejecución:</b> Se mostrará al usuario todos los CU que tiene asignado a su equipo. Del listado se debe permitir, filtrar por criterios, ver detalles del manual seleccionado y modificar los datos que necesite.	
<b>Resultado Esperado:</b> Se le mostrará el listado de CU al usuario.	
<b>Evaluación de la Prueba:</b> Prueba satisfactoria.	

Caso de Prueba de Aceptación	
<b>Código:</b> CP6_1	<b>Historia de Usuario:</b> 6
<b>Descripción:</b> Modificar los datos de un CU.	
<b>Condiciones de Ejecución:</b> El usuario debe estar autenticado y tener permiso para realizar esta acción en el sistema. Se debe seleccionar previamente el CU que se desea modificar.	
<b>Entrada/Pasos de Ejecución:</b> Se accede a modificar los datos deseados. Si se introducen datos inválidos o se deja algún campo vacío se debe mostrar un mensaje de error.	
<b>Resultado Esperado:</b> Se actualizan los datos del CU en el sistema.	
<b>Evaluación de la Prueba:</b> Prueba satisfactoria.	

Caso de Prueba de Aceptación	
<b>Código:</b> CP7_1	<b>Historia de Usuario:</b> 7
<b>Descripción:</b> Eliminar CU.	
<b>Condiciones de Ejecución:</b> El usuario debe estar autenticado y tener permiso para realizar esta acción en el sistema. Se debe seleccionar previamente el CU que se desea eliminar. El usuario debe tener permiso para realizar esta acción en el sistema.	
<b>Entrada/Pasos de Ejecución:</b> Se accede a eliminar el CU seleccionado, permitiendo confirmar la opción de eliminar el elemento. Una vez eliminado se muestra un mensaje de información al usuario.	
<b>Resultado Esperado:</b> Se elimina el CU seleccionado, del sistema.	
<b>Evaluación de la Prueba:</b> Prueba satisfactoria.	

Caso de Prueba de Aceptación	
<b>Código:</b> CP8_1	<b>Historia de Usuario:</b> 8
<b>Descripción:</b> Filtrar el listado de CU.	
<b>Condiciones de Ejecución:</b> El usuario debe estar autenticado y tener un listado de CU, asignado a su módulo.	
<b>Entrada/Pasos de Ejecución:</b> Se accede a la opción de filtrar el listado de CU mostrados al usuario por varios criterios de búsqueda. Si se introducen datos inválidos o se deja algún campo vacío se debe mostrar un mensaje de error.	
<b>Resultado Esperado:</b> El sistema muestra los CU coincidentes con los criterios de búsqueda especificados.	
<b>Evaluación de la Prueba:</b> Prueba satisfactoria.	

<b>Caso de Prueba de Aceptación</b>	
<b>Código:</b> CP9_1	<b>Historia de Usuario:</b> 9
<b>Descripción:</b> Cambiar estado de CU.	
<b>Condiciones de Ejecución:</b> El usuario debe estar autenticado y tener un listado de CU asignado a su módulo.	
<b>Entrada/Pasos de Ejecución:</b> Del listado de CU mostrados se debe acceder a la opción de modificar el estado del CU seleccionado. Una vez cambiado su estado se deben mostrar en pantalla todos los datos del CU.	
<b>Resultado Esperado:</b> El estado del CU es actualizado en el sistema.	
<b>Evaluación de la Prueba:</b> Prueba satisfactoria.	

<b>Caso de Prueba de Aceptación</b>	
<b>Código:</b> CP10_1	<b>Historia de Usuario:</b> 10
<b>Descripción:</b> Consultar CU.	
<b>Condiciones de Ejecución:</b> El usuario debe estar autenticado.	
<b>Entrada/Pasos de Ejecución:</b> Se debe acceder a la opción Consultas en el menú principal y luego introducir los datos de búsqueda. Si se introducen datos inválidos o se deja algún campo vacío se debe mostrar un mensaje de error. Una vez consultado el CU deseado se debe permitir ver sus datos.	
<b>Resultado Esperado:</b> Se muestran los CU coincidentes con los criterios de búsqueda seleccionados.	
<b>Evaluación de la Prueba:</b> Prueba satisfactoria.	

<b>Caso de Prueba de Aceptación</b>	
<b>Código:</b> CP11_1	<b>Historia de Usuario:</b> 11
<b>Descripción:</b> Incluir Manual de Usuario.	
<b>Condiciones de Ejecución:</b> El usuario debe estar autenticado y tener permiso para realizar esta acción en el sistema.	
<b>Entrada/Pasos de Ejecución:</b> Se debe acceder a la opción de registrar un nuevo Manual de Usuario en el sistema. Si se introducen datos inválidos o se deja algún campo vacío se debe mostrar un mensaje de error.	
<b>Resultado Esperado:</b> Se incluye los datos del Manual de Usuario satisfactoriamente en el sistema.	
<b>Evaluación de la Prueba:</b> Prueba satisfactoria.	

<b>Caso de Prueba de Aceptación</b>	
<b>Código:</b> CP12_1	<b>Historia de Usuario:</b> 12
<b>Descripción:</b> Mostrar listado de Manuales de Usuario.	
<b>Condiciones de Ejecución:</b> El usuario debe estar autenticado	
<b>Entrada/Pasos de Ejecución:</b> Se mostrará al usuario todos los Manuales de Usuario que tiene asignado a su equipo. Del listado se debe permitir, filtrar por criterios, ver detalles del manual seleccionado y modificar los datos que necesite. Sólo podrá modificar los datos, si el usuario autenticado tiene permiso para realizar esta acción en el sistema.	
<b>Resultado Esperado:</b> Se le mostrará el listado de manuales al usuario.	
<b>Evaluación de la Prueba:</b> Prueba satisfactoria.	



Caso de Prueba de Aceptación	
<b>Código:</b> CP_13	<b>Historia de Usuario:</b> 13
<b>Descripción:</b> Modificar Manual de Usuario.	
<b>Condiciones de Ejecución:</b> El usuario debe estar autenticado y tener permiso para realizar esta acción en el sistema. Se debe seleccionar previamente el manual que se desea modificar.	
<b>Entrada/Pasos de Ejecución:</b> Se accede a modificar los datos deseados del manual seleccionado. Si se introducen datos inválidos o se deja algún campo vacío se debe mostrar un mensaje de error.	
<b>Resultado Esperado:</b> Se actualizan en el sistema los datos modificados del manual seleccionado.	
<b>Evaluación de la Prueba:</b> Prueba satisfactoria.	

Caso de Prueba de Aceptación	
<b>Código:</b> CP_14	<b>Historia de Usuario:</b> 14
<b>Descripción:</b> Eliminar Manual de Usuario.	
<b>Condiciones de Ejecución:</b> El usuario debe estar autenticado y tener permiso para realizar esta acción en el sistema. Se debe seleccionar previamente el manual que se desea eliminar. El usuario debe tener permiso para realizar esta acción en el sistema.	
<b>Entrada/Pasos de Ejecución:</b> Se accede a eliminar el manual seleccionado, permitiendo confirmar la opción de eliminar el elemento. Una vez eliminado se muestra un mensaje de información al usuario.	
<b>Resultado Esperado:</b> Se elimina del sistema el manual seleccionado.	
<b>Evaluación de la Prueba:</b> Prueba satisfactoria.	

Caso de Prueba de Aceptación	
<b>Código:</b> CP15_1	<b>Historia de Usuario:</b> 15
<b>Descripción:</b> Filtrar listado de manuales de usuario.	
<b>Condiciones de Ejecución:</b> El usuario debe estar autenticado y tener un listado de manuales, asignado a su módulo.	
<b>Entrada/Pasos de Ejecución:</b> Se accede a la opción de filtrar el listado de manuales de usuario mostrados al usuario por varios criterios de búsqueda. Si se introducen datos inválidos o se deja algún campo vacío se debe mostrar un mensaje de error.	
<b>Resultado Esperado:</b> Se muestran los CU coincidentes con los criterios de búsqueda seleccionados.	
<b>Evaluación de la Prueba:</b> Prueba satisfactoria.	

Caso de Prueba de Aceptación	
<b>Código:</b> CP16_1	<b>Historia de Usuario:</b> 16
<b>Descripción:</b> Cambiar estado de Manual de Usuario.	
<b>Condiciones de Ejecución:</b> El usuario debe estar autenticado y tener un listado de manuales asignado a su módulo.	
<b>Entrada/Pasos de Ejecución:</b> Del listado de manuales de usuario mostrados se debe acceder a la opción de modificar el estado del manual seleccionado. Una vez cambiado su estado se deben mostrar en pantalla todos los datos del mismo.	
<b>Resultado Esperado:</b> Se actualiza en el sistema el estado del manual seleccionado.	
<b>Evaluación de la Prueba:</b> Prueba satisfactoria.	

<b>Caso de Prueba de Aceptación</b>	
<b>Código:</b> CP17_1	<b>Historia de Usuario:</b> 17
<b>Descripción:</b> Consultar manuales de usuario	
<b>Condiciones de Ejecución:</b> El usuario debe estar autenticado.	
<b>Entrada/Pasos de Ejecución:</b> Se debe acceder a la opción Consultas en el menú principal y luego introducir los datos de búsqueda. Si se introducen datos inválidos o se deja algún campo vacío se debe mostrar un mensaje de error. Una vez consultado el manual deseado se debe permitir ver sus datos.	
<b>Resultado Esperado:</b> Se muestran los manuales de usuario coincidentes con los criterios de búsqueda seleccionados.	
<b>Evaluación de la Prueba:</b> Prueba satisfactoria.	

<b>Caso de Prueba de Aceptación</b>	
<b>Código:</b> CP18_1	<b>Historia de Usuario:</b> 18
<b>Descripción:</b> Exportar a PDF.	
<b>Condiciones de Ejecución:</b> El usuario debe estar autenticado	
<b>Entrada/Pasos de Ejecución:</b> Se permitirá exportar a formato PDF el listado de CU o manuales mostrados en pantalla.	
<b>Resultado Esperado:</b> Se muestra en formato PDF el listado de CU o manuales de usuario.	
<b>Evaluación de la Prueba:</b> Prueba satisfactoria.	

<b>Caso de Prueba de Aceptación</b>	
<b>Código:</b> CP12_1	<b>Historia de Usuario:</b> 12
<b>Descripción:</b> Mostrar listado de Manuales de Usuario.	
<b>Condiciones de Ejecución:</b> El usuario debe estar autenticado	
<b>Entrada/Pasos de Ejecución:</b> Se mostrará al usuario todos los Manuales de Usuario que tiene asignado a su equipo. Del listado se debe permitir, filtrar por criterios, ver detalles del manual seleccionado y modificar los datos que necesite. Sólo podrá modificar los datos, si el usuario autenticado tiene permiso para realizar esta acción en el sistema.	
<b>Resultado Esperado:</b> Se le mostrará el listado de manuales al usuario.	
<b>Evaluación de la Prueba:</b> Prueba satisfactoria.	

<b>Caso de Prueba de Aceptación</b>	
<b>Código:</b> CP5_1	<b>Historia de Usuario:</b> 15
<b>Descripción:</b> Mostrar listado de Casos de Uso.	
<b>Condiciones de Ejecución:</b> El usuario debe estar autenticado	
<b>Entrada/Pasos de Ejecución:</b> Se mostrará al usuario todos los CU que tiene asignado a su equipo. Del listado se debe permitir, filtrar por criterios, ver detalles del manual seleccionado y modificar los datos que necesite.	
<b>Resultado Esperado:</b> Se le mostrará el listado de CU al usuario.	
<b>Evaluación de la Prueba:</b> Prueba satisfactoria.	

## Anexo 6. Acta de Aceptación del producto de software realizado.

### Acta de Aceptación

**Producto:** Sistema de Control de Estado de las Especificaciones de Casos de Uso y Manuales de Usuario del SIIPOL.

**Categoría del Producto:** Aplicación Web.

**Fecha de conciliación:** 30 de Abril de 2010

**Involucrados en el proceso:**

- **Por la parte del Cliente:** Miguel Ángel Monagas Reyes.
- **Por la parte suministradora:** Ivonne Isabel Guerra Anaya

Alexey Casanova Álvarez

- **Observador independiente (Calidad Interna CICPC):** María A. Rubalcaba Betancourt.

**Observaciones del proceso:**

Durante el proceso de Pruebas se detectaron un conjunto de defectos, que quedaron registrados adecuadamente en el correspondiente documento de No Conformidades detectadas, con sus respectivas observaciones. Teniendo en cuenta que las No Conformidades han sido debidamente resueltas por el Equipo de Desarrollo, y validada la eficacia de la corrección por los Clientes, se ha tomado el acuerdo del producto: "Sistema de Control de Estado de las Especificaciones de Casos de Uso y Manuales de Usuario del SIIPOL"

Para que conste la Aceptación de los resultados de las Pruebas, y por tanto la Aceptación del producto, dando fe del acuerdo firman la presente, los principales representantes de las Partes.

  
Ivonne Isabel Guerra Anaya

  
Alexey Casanova Alvarez

  
Miguel A. Monagas Reyes  
(Representante del Cliente)

  
María A. Rubalcaba Betancourt  
(Calidad Interna CICPC)

---

**Anexo 6. Encuesta realizada a los usuarios.****Encuesta a usuarios del sistema****Nombre del Encuestado:****Rol que desempeña:****Años de Experiencia en el proyecto:**

1. **¿Considera usted, como usuario, que la herramienta es compleja de utilizar?**

Si\_\_\_\_ No\_\_\_\_

2. **¿Se puede considerar útil la herramienta para el control de los estados de los casos de uso y manuales de usuarios dentro del proceso de desarrollo?**

Si\_\_\_\_ No\_\_\_\_

¿Por qué?

---

---

---

3. **¿El acceso, navegación y consulta dentro de la aplicación se realizan de manera rápida y sencilla, lo que permite disponer de información oportuna?**

Si\_\_\_\_ No\_\_\_\_ Muy poco\_\_\_\_

4. **¿El sistema le serviría como una herramienta de trabajo para facilitar el desarrollo de sus actividades como jefe de equipo controlando el trabajo realizado?**

Si\_\_\_\_ No\_\_\_\_ Muy poco\_\_\_\_

5. **¿La herramienta le serviría para centralizar y organizar la información referente al trabajo que usted controla?**

Si\_\_\_\_ No\_\_\_\_ Muy poco\_\_\_\_

6. **¿Considera que sea un mecanismo eficiente para comunicarle usted, cuáles son los casos de usos o manuales de usuario con los que ya pueden trabajar?**

Si \_\_\_\_ No \_\_\_\_

¿Por qué no?

---

---

---

7. **¿Considera que el uso de la aplicación ahorra tiempo a la hora de que el líder del proyecto conozca las actividades realizadas durante el día a partir los casos de usos y manuales de usuario?**

Sí \_\_\_\_ No \_\_\_\_

¿Por qué?

---

---

---

8. **¿Piensa que la herramienta controle eficientemente el avance del proyecto a partir de los estados que tengan los casos de uso o manuales de usuario, dentro del desarrollo del software?**

Sí \_\_\_\_ No \_\_\_\_

¿Por qué?

---

---

---

9. **Considera recomendable el uso de la herramienta en el proyecto.**

Si \_\_\_\_ No \_\_\_\_

**Evalúe del 1 al 5 los criterios valorados en las preguntas anteriores, tomando el grado de adecuación de los mismos a las necesidades puntuales de su trabajo en el proyecto.**

Pregunta 1\_\_\_\_

Pregunta 2\_\_\_\_

Pregunta 3\_\_\_\_

Pregunta 4\_\_\_\_

Pregunta 5\_\_\_\_

Pregunta 6\_\_\_\_

Pregunta 7\_\_\_\_

Pregunta 8\_\_\_\_

Pregunta 9\_\_\_\_

Pregunta 10\_\_\_\_