

Universidad de las Ciencias Informáticas  
Facultad 5



**“SELECCIÓN Y PERSONALIZACIÓN DE UNA HERRAMIENTA  
PARA LA GESTIÓN DE PROYECTOS DE SOFTWARE EN LA  
UNIVERSIDAD DE LAS CIENCIAS INFORMÁTICAS”**

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

**Autor(es):** Yaillet La Rosa Pérez de Alejo

Alejandro Fernández Díaz

**Tutor(es):** Msc. Anisleiby Fernández Hernández

Ing. Henrik Pestano Pino

**Ciudad de La Habana, junio 2010**

“Año del 50 Aniversario del Triunfo de la Revolución”

*“El propósito fundamental es lograr un centro de excelencia para la formación masiva de profesionales de nivel superior, lo que deberá alcanzarse con la ejecución de ambiciosos programas curriculares y de producción y con la aplicación de las más modernas tecnologías en la docencia”.*

*Castro-Díaz-Balart*

## DECLARACIÓN DE AUTORÍA

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

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

Autores:

Yaillet La Rosa Pérez de Alejo

Alejandro Fernández Díaz

\_\_\_\_\_

\_\_\_\_\_

Tutor:

Tutor:

Msc.Anisleiby Fernández

Henrik Pestano Pino

\_\_\_\_\_

\_\_\_\_\_

## DATOS DE CONTACTO

---

**Nombre y Apellidos:** Msc. Anisleiby Fernández Hernández

**E-mail:** ani@uci.cu

**Nombre y Apellidos:** Ing. Henrik Pestano Pino

**E-mail:** hpestano@uci.cu

### Agradecimientos

*A mi papá por ser padre y madre, por demostrarme que siempre se puede un poquito más, gracias papá porque sin tu apoyo hoy no estuviera aquí. A mi hermano, que es la persona que más amo, Tata yo espero lo mejor de ti. A mi mamá Mayelín por estar conmigo en todo momento. A mi tía, gracias por tu apoyo incondicional. A mi mamá, porque a pesar de no estar a mi lado en los momentos difíciles yo la siento siempre cerca de mí. A mi prima que más que prima es mi hermana. A mi mami querida, Nelia no sabes lo importante que eres en mi vida. A mi abuela Fe, gracias abuela porque cuando estuve lejos de casa me hiciste sentir como en ella. A toda la gente linda que conocí en el transcurso de estos años, personas que nunca voy a olvidar porque se convirtieron en parte de mi vida, gracias Yoisy por tener tanta paciencia conmigo, gracias Lida porque aunque no te encuentres hoy aquí quiero que sepas que aprendí mucho de ti como persona. A mis grandes compañeras Mirtha, Lisandra, Yamila, Mayliubi y Leydis. A mi hermana Papita, te conocí de cerca y eres la persona más linda que he conocido en todos estos años. A mis hermanitos Yasin y Luisdey. A todos los que contribuyeron a realizar este sueño.*

*Yaillet*

## AGRADECIMIENTOS

---

*A mi familia por a verme apoyado siempre en los momentos difíciles.*

*Alejandro*

**Dedicatoria**

*A mi papá, gracias por tu confianza en mí.*

*A mi hermano, eres lo más hermoso en mi vida.*

*A mis tres madres, las quiero mucho.*

*A Nelia, te amo.*

*A toda mi familia.*

*Yaijet*

*A mi mamá, mi papá y mi novia.*

*A toda mi familia.*

*Alejandro*

### **Resumen**

La presente investigación pretende analizar todo lo referente a los Sistemas de Gestión de Proyectos de Software y a las herramientas de desarrollo colaborativo. Para lograr dicho objetivo se realizó una caracterización de cada uno de los conceptos que forman parte del tema de los Sistemas de Gestión de Proyectos de Software, y la descripción del proceso de desarrollo del software, con las diferentes actividades que lo integran, definiciones, antecedentes y las metodologías fundamentales estudiadas en esta disciplina, también se incluyó el proceso de selección de una herramienta de desarrollo colaborativo candidata para llevar a cabo la Gestión de Proyectos de Software. El análisis realizado a estos aspectos busca obtener una propuesta para implantar la herramienta que más se ajuste con los procesos tratados hoy en día en la UCI. También es realizada una validación de la herramienta que se propone, a partir de los resultados obtenidos de las estadísticas llevadas a cabo para evidenciar su puesta en uso en la universidad.



## Índice

NOVEDAD .....	7
ALCANCE DE LA INVESTIGACIÓN.....	7
<b>CAPÍTULO 1. SISTEMAS DE GESTIÓN DE PROYECTOS. ....</b>	<b>10</b>
1.1 SISTEMAS DE INFORMACIÓN. ....	10
1.1.1 Sistemas de Información. Actividades.....	11
1.1.2 Sistemas de Información. Características. ....	13
1.1.3 Sistemas de Información. Metodologías.....	14
1.2 SISTEMAS DE GESTIÓN DE INFORMACIÓN. CARACTERIZACIÓN. ....	17
1.3 GESTIÓN DE PROYECTOS.....	19
1.3.1 Definiciones. ....	19
1.3.2 Gestión de Proyectos. Antecedentes .....	20
1.3.3 Gestión de Proyectos. Características. ....	20
1.3.4 Sistemas de Gestión de Proyecto. ....	22
<b>CAPÍTULO 2. ELECCIÓN DE LA HERRAMIENTA PARA EL DESARROLLO DE LA GESTIÓN DE PROYECTOS EN LA UCI. ....</b>	<b>24</b>
2.1 ESTUDIO COMPARATIVO DE LAS HERRAMIENTAS PARA EL DESARROLLO DE SISTEMAS DE GESTIÓN DE PROYECTOS DE SOFTWARE. ....	24
2.2 HERRAMIENTAS DE DESARROLLO COLABORATIVO. ....	24
2.3 COMO LLEVAR EL PROCESO DE ELECCIÓN DE UNA HERRAMIENTA. ....	25
2.4 HERRAMIENTAS DE GESTIÓN DE PROYECTO UTILIZADAS EN LA UCI. ....	28
2.5 RESULTADOS DEL ESTUDIO COMPARATIVO. ....	36
2.6 PROPUESTA DE HERRAMIENTA A UTILIZAR. ....	39
2.6.1 Arquitectura de RedMine. ....	39
2.6.2 Arquitectura Modelo-Vista-Controlador (MVC) de Rails. ....	40
2.7 DESCRIPCIÓN DE PLUG-INS (FUNCIONALIDADES DEL REDMINE). ....	43
2.7.1 Reportes. ....	43
2.7.2 Riesgos.....	45
2.7.3 Asistencia .....	46
<b>CAPÍTULO 3. VALIDACIÓN DE LA HERRAMIENTA Y PUESTA EN PRÁCTICA. ....</b>	<b>47</b>
3.1 NO CONFORMIDADES O DIFICULTADES QUE PRESENTA LA HERRAMIENTA.....	48
3.2 USUARIOS ACTIVOS.....	49

3.3 AVANCES DE LA HERRAMIENTA.....	50
<b>CONCLUSIONES.....</b>	<b>52</b>
<b>RECOMENDACIONES.....</b>	<b>53</b>
<b>BIBLIOGRAFÍA REFERENCIADA .....</b>	<b>54</b>
<b>ÍNDICE DE FIGURAS Y TABLAS .....</b>	<b>57</b>
<b>GLOSARIO DE TÉRMINOS.....</b>	<b>65</b>

## Introducción

En Cuba se han llevado a cabo numerosos proyectos en cuanto al desarrollo de software. La Universidad de las Ciencias Informáticas (UCI) constituye un ejemplo de ello. Surgida en el 2002, no fue concebida como una universidad tradicional, sino como un centro de estudios productivos además de su proceso docente-investigativo, llevando a cabo el principio de Investigación-Formación-Producción. En este contexto la producción de software es llevada a cabo desde la vinculación estudio-trabajo, como modelo de formación. Esta nueva concepción permite que los estudiantes estén constantemente involucrados en la producción, llevando a cabo un proceso de formación desde la misma. ([Fernández, 2009](#)).

Constituye además un puente importante para la empresa comercializadora de software, a partir de intercambios de cooperación y colaboración que se establece con otros países, proporcionándole al país mayores beneficios y oportunidades económicas en el desarrollo de software a la medida. Actualmente la UCI es considerada uno de los centros de mayor producción de software en Cuba, a pesar de los varios intentos en la universidad de mantener este campo de estudio aún adolece de buenas prácticas que permitan llevarlo a cabo de manera efectiva.

La Gestión de Proyectos es de vital importancia para la evolución del producto. En ella se aplican e integran los procesos de dirección de proyectos de inicio, planificación, ejecución, seguimiento, control y cierre. ([Project Management Institute, 2004](#)).

"En los inicios de un nuevo milenio, en un entorno agresivo, globalizado y dinámicamente cambiante, cuando se hace referencia cada vez más en la Gestión de Proyecto, la Ciencia Cubana tiene también aportes, soluciones y experiencias que brindar para alcanzar el imprescindible desarrollo sostenible al que aspiran los pueblos del tercer mundo". ([Fernández, 2009](#)).

Esto significa que la lucha por el desarrollo hay que librarla con la convicción de que cada solución está en la búsqueda propia de resultados eficaces y eficientes, así como en el desarrollo de una cultura de gestión y planificación en función de las profundas transformaciones económicas, cuyo hilo conductor es la investigación y la innovación.

En la Universidad de las Ciencias Informáticas desde su surgimiento le preocupó si los productos y/o servicios de software que se producen se ofrecen con la calidad requerida. Por lo que la Gestión de Proyecto se ha convertido en el elemento clave de la evolución de los sistemas y productos informáticos. En los pasados 3 años, la producción de software en la UCI ha pasado de ser una resolución de problemas especializada y una herramienta de análisis de información, a ser una industria por sí misma.

La Gestión de Proyectos de Software adquiere mayor importancia aún en el aumento de la complejidad de un proyecto de software donde se quiera lograr como resultado un producto de alto nivel de complejidad. En este caso las restricciones de calidad y la falta de planificación en cada una de las etapas provocarán el aumento de los costos, el tiempo de desarrollo y el aumento de los riesgos, lo cual pone en peligro el desarrollo del proyecto. ([Fernández, 2009](#)).

En los proyectos de desarrollo de software la eficaz Gestión de Proyecto posibilita conocer los recursos con que se cuenta para desarrollar el mismo. Permite optimizarlos de la mejor manera posible, así como conocer las limitaciones a tener en cuenta. También los factores críticos como son el tiempo y el presupuesto. Este conocimiento permite perfeccionar el trabajo y adecuarlo a los requisitos más específicos o más relevantes del proyecto. ([PMBOK, 2004](#)).

Los Sistemas de Gestión de Proyectos son los encargados de llevar a buen fin la Gestión de Proyectos, es necesario que la organización cuente con ciertos elementos claves. El conjunto de esos elementos constituye un Sistema de Gestión de Proyectos en el que cada elemento individual se denominaría un subsistema. Dado que todos los sistemas se componen de entradas, salidas y un proceso que transforme las entradas en salidas, lo mismo puede decirse de cada uno de los componentes de un Sistema de Gestión de Proyectos. ([Stewart & Gess, 2009](#)).

El primer criterio para definir una herramienta de Gestión de Proyectos o cualquier tipo de herramientas es **saber para qué se supone que debe servir**. Hay que medirla con el proceso a ver si lo soporta y no al revés como hacen muchos, que intentan ajustar el proceso a la herramienta; esto sólo genera el fracaso.

A modo general son estas las principales actividades que debe permitir una herramienta de Gestión de Proyectos. ([Stuwart & Gess, 2009](#)):

1. **Sistemas de seguimiento (tracker) genéricos.** Que permita definir tracker además de los básicos: Actividades, Errores, Riesgos, Incidencias, etc.... como convenga según el proceso de gestión. Por ejemplo: si se quiere medir recursos y adicionar campos totalmente distintos como nombre, recurso, tipo, responsable, costo, etc.
2. **Definir flujos de trabajo.** Importante para el cambio de status en actividades, errores, etc.
3. **Dejar evidencia de las trazas.** Importante poder tener el control de las trazas que se van generando entre actividades, errores, riesgos, etc., y entre estos también ej.: trazas de errores-actividades, entre otros.
4. **Monitoreo global de la organización y de proyectos en específico.** Es necesario que en todo momento se pueda saber el estado de la organización como un todo y además de algún proyecto en específico.
5. **Monitoreo del avance real del proyecto u organización.** Mediante métodos tradicionales como el cálculo de Valor Ganado.
6. **Asignación de actividades a rol o grupo de personas.** Esto se refiere a que debería ser posible, a pesar de que es mucho mejor luchar por la atipicidad de las actividades (esto es una actividad por persona), debe existir la posibilidad de asignarla a más de uno; por ejemplo las actividades que no pueden ser realizadas por una persona como la captura de requisitos, el análisis o la revisión de la arquitectura, donde el consenso entre el equipo al que se asignó la actividad es lo más importante.
7. **Que permita modelar la organización.** Ej. Gerencia -> Facultad -> área temática -> Proyecto-> Subproyecto -> Módulo -> documento, tarea, fichero de código, actividad, errores, riesgos, etc.
8. **Funcionalidades gráficas (Gantt, PERT).**

9. **Funciones colaborativas (email, RSS, foros, calendario común, etc.).**
10. **Sistema de alertas.** Notificaciones o avisos a los usuarios, o a los responsables de la gestión cuando se ejecute determinado evento o cuando se requiera.
11. **Gestión documental.** Este punto es realmente importante en organizaciones complejas, porque no siempre un simple control de versiones y un acceso seguro es suficiente para gestionar la documentación de un proyecto. Más aún si entendemos que la documentación por sí misma no sirve para nada, sino que es uno de los pilares de la gestión del conocimiento.
12. **Gestión del portafolio de proyectos.** Ya que los proyectos forman parte de un conjunto más amplio y si bien tienen entidad propia, han de ser contemplados como parte de un grupo aún mayor, donde las interdependencias entre los proyectos y la alineación de los mismos a la visión estratégica y a las capacidades de la organización es fundamental.

Muchas herramientas existen en el mundo de la informática ideadas para la planificación, monitoreo y control de proyectos. Existen las que gestionan un sólo proyecto, las que son multiproyectos, las colaborativas, etc. Realmente todo un universo de herramientas que tienen similares funcionalidades, siendo realmente un quebradero de cabeza definir cuál o cuáles son las que realmente se necesitan para llevar a cabo el proceso de desarrollo de alguna empresa u organización determinada.

Las herramientas para desarrollo colaborativo pueden ser utilizadas de forma individual o combinada, de acuerdo a la estrategia que se proponga el grupo de trabajo que las utiliza. Estas herramientas son los medios que permiten acceder a ciertos servicios que facilitan a los usuarios comunicarse y trabajar en conjunto sin importar que estén reunidos en un mismo lugar. Normalmente con ellos se puede compartir información en determinados formatos (audio, texto, video), y en algunos casos producir conjuntamente nuevos materiales productos de la colaboración.

En un estudio realizado en conjunto con la Dirección Técnica en el año 2007-2008 donde la universidad contaba con 150 proyectos, de los que se tomaron de 96 a 100 para realizarles entrevistas a sus líderes, que representan un 64% aproximadamente del total de proyectos existentes.

Entre los problemas que más afectan directamente en la no realización o fracaso de los mismos se encuentran la mala gestión del equipo de trabajo así también como la de sus recursos, además de imperfecciones en la planificación exacta de los proyectos, lo que provoca muchas veces atraso en el cumplimiento de los cronogramas. Existen deficiencias en la estimación exacta de los costos de ejecución, lo que puede llegar a limitar el aporte económico a obtenerse. En algunos casos no se lleva a cabo un control y planificación eficiente de la calidad en los proyectos. La planificación, seguimiento y control de los riesgos y su mitigación muchas veces no es la más adecuada.

Lo anteriormente dicho crea una **situación problemática** sobre la necesidad de elegir y personalizar una herramienta para llevar a cabo un adecuado proceso de gestión de software, que logre agrupar las mayores funcionalidades posibles de manera efectiva en la Universidad de las Ciencias Informáticas.

Basado en las motivaciones anteriormente planteadas se tiene como **problema científico**: ¿Cómo resolver la necesidad de seleccionar una herramienta para la Gestión de Proyectos de Software?

El **objeto de estudio** el dominio de la Gestión de Proyectos y el **campo de acción** el proceso de selección de una herramienta.

De acuerdo con todo lo expuesto se trazó como **objetivo general** “Elección y personalización de una herramienta para la Gestión de Proyectos de Software en la Universidad de las Ciencia Informáticas”.

Para una mejor orientación de la investigación se trazaron los siguientes **objetivos específicos**.

1. Abordar sobre las principales tendencias y aplicaciones que existen sobre los sistemas de gestión relacionados con el desarrollo de proyectos de software.
2. Evidenciar a través de un estudio exploratorio cómo se lleva a cabo la Gestión de Proyectos de Software en la UCI.
3. Realizar un estudio comparativo de las posibles herramientas colaborativas existentes que soportan sistemas de gestión para llevar a cabo la Gestión de Proyectos de Software.

4. Realizar la propuesta del sistema para la Gestión de Proyectos de Software en la UCI.
5. Validar la propuesta del sistema.

### **Tareas por objetivos:**

- 1- Caracterización de las principales tendencias y aplicaciones de los sistemas de gestión.
- 2- Aplicación de entrevistas que permitan conocer el proceso de desarrollo de proyectos de software en la UCI.
- 3- Estudio estadístico de los proyectos en la UCI que permite conocer cómo se realiza la Gestión de Proyecto.
- 4- Descripción de las herramientas colaborativas empleadas en la UCI para definir cuál de ellas es la más óptima.
- 5- Conceptualización de la propuesta de solución.

### **Resultados esperados:**

Los posibles resultados que se deben obtener de este sistema de Gestión de Proyectos Software dependen del esfuerzo, dedicación y seriedad del trabajo, de la calidad de la información a investigar y su validez. El sistema debe garantizar que los flujos de información en el centro sean los más exactos posibles de manera que la información que salga del sistema sea correcta y segura para lograr un verdadero control de todos los procesos que se quieren llevar a cabo, pero de manera general la investigación realizada debe:

- 1- Destacar las principales tendencias que existen hoy día referentes a los Sistemas de Gestión Proyectos Informáticos.
- 2- Dar a conocer la situación actual en la que se encuentra la Gestión de Proyectos de Software en la UCI.



3- Proponer una herramienta que permita llevar a cabo la Gestión de Proyectos de Software de manera efectiva en la UCI.

4-Consultar la documentación y los manuales, puestos a su disposición para permitir el consumo del servicio por parte de los jefes de proyecto.

5- Lograr que los proyectos de la universidad sean gestionados de manera homogénea, de esta forma, todos los jefes de proyectos sabrán lo que tienen que hacer y cómo.

### **Novedad**

Este estudio permitirá tener una visión más completa acerca de la Gestión de Proyectos de Software especialmente en la Universidad de las Ciencias Informáticas. Desde el análisis de las características de los Sistemas de Gestión de Proyectos, se llega a conclusiones, que vistas desde este enfoque permiten al menos un primer acercamiento al estudio del arte de la Gestión de Proyectos de Software. Este estudio también aportará los elementos esenciales para la correcta selección de una herramienta de desarrollo colaborativo para solucionar los problemas más frecuentes que presenta la Gestión de Proyecto.

### **Alcance de la Investigación**

Analizar el estado del arte que conforman los Sistemas de Gestión de Proyectos de Software integrado por las diferentes disciplinas y características que lo integran. Evidenciar cómo se comporta la Gestión de Proyecto en la Universidad de las Ciencias Informáticas. Realizar una aproximación estadística a través de los resultados obtenidos en el análisis del estudio realizado por la Dirección Técnica en el período de 2007-2008, usando preferentemente indicadores simples de actividad.

Para la realización de la propuesta de solución del Sistema de Gestión de Proyectos de Software, se plantean los siguientes **métodos**:

Es necesario ante todo un análisis previo de las necesidades de información del centro con respecto al tema a tratar, un diagnóstico de la situación real que permita conocer los recursos de información disponibles y los que faltan, para dicho trabajo se utilizaron los métodos científicos de la investigación, ejemplo el método histórico-lógico empleado para realizar un estudio de los principales postulados teóricos sobre el tema, partiendo

desde el surgimiento y análisis de los antecedentes hasta las características principales, el método analítico-sintético, fue utilizado en el estudio de las teorías y documentos para la extracción de elementos importantes, también se hace uso del inductivo-deductivo para llegar a un consenso de la investigación además de métodos para recopilar información como lo fue la entrevista no estandarizada, de esta forma logrando comunicación y adquiriendo conocimientos respecto al tema.

**Entrevista no estandarizada:** Es una técnica de entrevista en la cual no se sigue una guía previamente elaborada, sino, que los participantes respondan los puntos que el entrevistador tenga como objetivo conocer. De esta manera, pueden ir surgiendo nuevas ideas referentes a la cuestión y que no estaban contempladas por el entrevistador. Todas las opiniones del participante son tomadas en cuenta.

Los criterios que se tuvieron en cuenta para su selección fueron:

- Desarrollo de cronograma
- Gestionar tiempos muertos
- Gestión de proyecto
- Controlar cambios realizados a proyectos
- Aplicación de la herramienta
- Dominio de la gestión de proyecto
- Desarrollo de herramientas colaborativas
- Control de cambios en los proyectos
- Reportes de estado
- Gestionar riesgos
- Gestionar alcance
- Gestionar costo

## **Estructura Capitular**

### **Capítulo: 1** “Fundamentación Teórica”

Se expone todo lo referente al marco teórico conceptual de Sistemas de Gestión de Proyectos, conceptos fundamentales, antecedentes, características etc.

### **Capítulo: 2** “Propuesta Solución”

Se realiza un estudio comparativo de las diferentes herramientas colaborativas utilizadas en la universidad y se da a conocer la propuesta solución.

### **Capítulo: 3** “Validación de la Herramienta”

Se ofrece la efectividad de la herramienta elegida, estadísticas actuales de su utilización

## Capítulo 1. Sistemas de Gestión de Proyectos.

### 1.1 Sistemas de Información.

En sus inicios los Sistemas de Información fueron considerados como un elemento que podía proporcionar ahorro a las organizaciones, además de dar soportes a actividades operativas en la que la información constituía el principal elemento implicado. Ya en la década de los 90 las computadoras no eran solo herramientas para especialistas técnicos sino que se convirtieron en parte del ambiente de trabajo, utilizadas por personas de todo nivel de la organización, esto trajo consigo que las empresas tuvieran más información disponible, mediante esta situación se volvieron más complejos los problemas de administrar la información y los desafíos de hacer un uso adecuado de la información y la tecnología. A raíz de esto los problemas aumentaron en tamaño y complejidad por lo que los Sistemas de Información prestaron mayor atención al uso de las tecnologías de computación como un medio en la comunicación y la toma de decisiones en las organizaciones.

([Lesca,1986](#)) plantea que *“El Sistema de Información es la manera en que la organización almacena ordena y procesa los datos necesarios que tributan a la organización y estará en función de los diferentes procesos que en ella tiene lugar, interrelacionando a todos los departamentos entre sí y tributando en tiempo real y con objetividad al proceso de toma de decisiones de los directivos, es una herramienta más, que en el momento de la planeación estratégica, se puede utilizar eficientemente en función del control interno de la organización.”*

Por su parte ([Mengunzatto & Renau, 1995](#)), considera que un **Sistema de Información** es *“ un conjunto de elementos que interactúan entre sí con el fin de apoyar las actividades de una empresa o negocio. Lo esencial de todo Sistema de Información es que mediante él se va a proporcionar la información necesaria, en el momento oportuno y con la estructura adecuada a aquellos miembros que la requieran para diversos usos.”*

([Peña, 2006](#)) plantea que *“ Un Sistema de Información es un conjunto de elementos interrelacionados con el propósito de prestar atención a las demandas de información de una organización, para elevar el nivel de conocimientos que permitan un mejor apoyo a la toma de decisiones y desarrollo de acciones. ”*

Otros autores como ([Peralta, 2008](#)), de una manera más acertada define Sistema de Información como: *“conjunto de elementos que interactúan entre sí con el fin de apoyar las actividades de una empresa o negocio. Teniendo muy en cuenta el equipo computacional necesario para que el Sistema de Información pueda operar y el recurso humano que interactúa con el Sistema de Información, el cual está formado por las personas que utilizan el sistema. ”*

Estos sistemas regulan todo tipo de información con el objetivo de lograr cambios en la solución de problemas y en la toma de decisiones de empresas o negocios.

En teoría de sistemas un Sistema de Información es un sistema automatizado o manual que involucra personas, máquinas y/o métodos organizados de recolección, procesos, transmisión clasificar datos que divulguen información del usuario.

### **1.1.1 Sistemas de Información. Actividades.**

Un Sistema de Información realiza cuatro actividades básicas: **entrada, almacenamiento, procesamiento y salida de información** ([Peralta, 2008](#)).

**Entrada de información:** Es el proceso mediante el cual el Sistema de Información toma los datos que requiere para procesar la información. Las entradas pueden ser manuales o automáticas. Las manuales son aquellas que se proporcionan en forma directa por el usuario, mientras que las automáticas son datos o información que provienen o son tomados de otros sistemas o módulos. Esto último se denomina interfaces automáticas. Las unidades típicas de entrada de datos a las computadoras son las terminales, las cintas magnéticas, las unidades de diskette, los códigos de barras, los escáner, la voz, los monitores sensibles al tacto, el teclado y el mouse, entre otras.

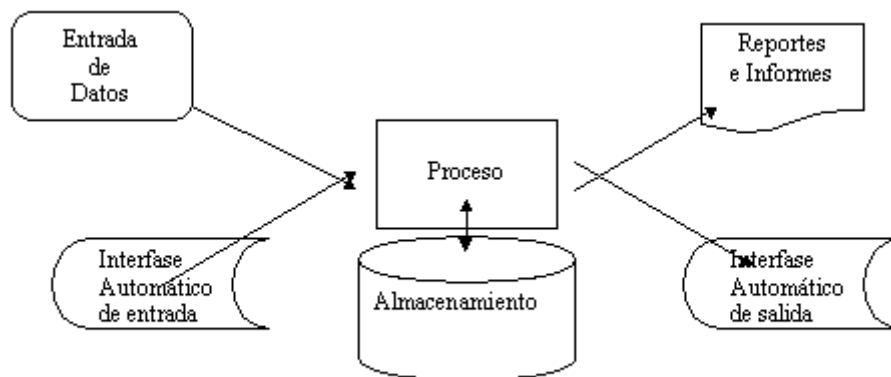
**Almacenamiento de información:** El almacenamiento es una de las actividades o capacidades más importantes que tiene una computadora, ya que a través de esta

propiedad el sistema puede recordar la información guardada en la sección o proceso anterior. Esta información suele ser almacenada en estructuras de información denominadas archivos. La unidad típica de almacenamiento son los discos magnéticos o discos duros, los discos flexibles o diskettes y los discos compactos (CD-ROM).

**Procesamiento de información:** Es la capacidad del Sistema de Información para efectuar cálculos de acuerdo con una secuencia de operaciones preestablecida. Estos cálculos pueden efectuarse con datos introducidos recientemente en el sistema o bien con datos que están almacenados. Esta característica de los sistemas permite la transformación de datos fuente en información que puede ser utilizada para la toma de decisiones, lo que hace posible, entre otras cosas, que un tomador de decisiones genere una proyección financiera a partir de los datos que contiene un estado de resultados o un balance general de un año base.

**Salida de información:** La salida es la capacidad de un Sistema de Información para sacar la información procesada o bien datos de entrada al exterior. Las unidades típicas de salida son las impresoras, terminales, diskettes, cintas magnéticas, la voz, los graficadores y los plotters. Es importante aclarar que la salida de un Sistema de Información puede constituir la entrada a otro Sistema de Información o módulo. En este caso, también existe una interface automática de salida.

En la siguiente figura se representa la relación de las actividades de estos sistemas de información.



**Figure 1 Actividades del Sistema de Información.**

### 1.1.2 Sistemas de Información. Características.

Algunas de las características de estos Sistemas de Información son imprescindibles para su buen funcionamiento, por ejemplo para que un Sistema de Información funcione al máximo debe cumplir al 100%, tener puntualidad o sea proporcionar y distribuir información confiable y actualizada a los usuarios que la necesiten y en el momento que la soliciten, una buena precisión a la hora de tomar las decisiones, consistencia, proporcionar confianza y solidez en la información a través de una buena recopilación de datos, debe ser capaz de brindar información completa en toda su totalidad y plenitud para la realización de un buen trabajo, si uno de estos elementos deja de cumplirse pues estaría en peligro lograr el objetivo del sistema.

La organización de los datos en información: Para que los datos sean significativos, deben tener un propósito. Los datos almacenados deben reflejar el propósito y el tipo de Sistema de Información. Las necesidades de datos para ser procesados y organizada antes de que sea la información. La organización de los datos más probable es que implican los procesos de clasificación y filtrado (la clasificación) antes de que pueda ser analizada y almacenada para su posterior recuperación. Los diccionarios de datos se utilizan para ayudar a organizar los datos.

Capacidad de analizar la información: Una vez que los datos se han convertido en la información que necesita para ser analizados para aprovechar al máximo la información almacenada. Análisis de bases de datos se realiza a través de las herramientas de consultas e informes.

Son complejos y grandes. La complejidad se asocia a la diversidad de componentes que los conforman y de las relaciones que están presentes en ellos. Por lo general, todo lo relativo a consideraciones sociales, económicas, políticas y cognitivas se consideran complejas. Estas mismas características hacen de otro aspecto vital lo sea su capacidad para responder a los cambios, adaptarse a su ambiente y mantener cierta estabilidad que les permita sobrevivir.

El proceso que se lleva a cabo en el sistema es dinámico porque cuando se envía (autor), un material para publicar, los encargados de revisar dicho documento al acceder al sistema se les muestra una lista con todos los trabajos pendientes y estos deciden cuales tienen los requisitos para su publicación.

Inmediatez de la publicación y edición de la información, con esto se gana agilidad para la divulgación del contenido en una institución.

### 1.1.3 Sistemas de Información. Metodologías.

Los Sistemas de Información basados en computadoras son los más usados en la actualidad debido a su uso para diversas finalidades como desde el procesamiento de las transacciones de una empresa hasta proveer de la información necesaria para decidir sobre asuntos comunes que se presentan con frecuencias, además de la asistencia brindada a los funcionarios de la organización. Teniendo en cuenta la importancia de estos sistemas, a la hora de comenzar su creación es preciso explorar detalladamente cada método existente para llevar a cabo su desarrollo teniendo en cuenta el más apropiado para la obtención del mayor beneficio para la empresa.

#### 1.1.3.1 Método del ciclo de vida de desarrollo de un sistema.

En la figura número 2 se representan las diferentes etapas por las que transcurre este método.



Figure 2 Etapas del ciclo de vida de los sistemas. (Ponjuan, 2004).

Este método incluye actividades de investigación preliminar, determinación de requerimientos, diseño del sistema, desarrollo de software, prueba de sistema e implantación. Características de esta aplicación los requerimientos del sistema de información deben ser predecibles. Manejable como proyecto. Requiere que los datos



se encuentren en archivos y bases de datos. Gran volumen de transacciones y requerimientos. Requiere de la validación de los datos de entradas. Tiempo de desarrollo largo. Desarrollo por equipo de trabajo. Adecuado para todo tipo de aplicaciones. Mayor utilidad como complemento de otros métodos de desarrollo. Presenta varias etapas ( [Ponjuan, 2004](#)):

**Estudio preliminar:** Esta primera etapa la llevan a cabo los analistas de los sistemas además de una representación de los usuarios. El trabajo del analista no es más observar en observar detenidamente lo que ocurre en la organización, porque muchas veces los requerimientos no están claramente establecidos, por lo que, el proyecto requerido debe examinarse para determinar precisamente lo que desea la organización

**Determinación de los requerimientos:** Los analistas utilizan una variedad de métodos a fin de recopilar los datos sobre una situación existente, como entrevistas cuestionarios, inspección de registros (revisión in situ) y observación. Básicamente se trata de analizar y definir las decisiones que se toman en la organización o departamento al que prestara apoyo el Sistema de Información, estudiar los procesos de toma de decisión (cómo se toman las decisiones y como deberían tomarse) y definir las necesidades de información para los mismos.

**Diseño:** Si con el análisis se persigue determinar qué debe hacer el nuevo sistema, el objetivo de la fase de diseño es establecer cómo debe hacerlo. Ello implica definir el hardware y software que utilizará el sistema, los flujos de datos, la estructura de las bases de datos, los procedimientos y los módulos de procesamiento o programas informáticos.

**Desarrollo:** En esta fase se confeccionan los programas y módulos diseñados en la etapa anterior, ponerlos a punto de ejecución y efectuar las pruebas planificadas.

**Prueba:** Prueba general a todos los procedimientos y programas que se han generado y le aporta al usuario experiencia y entrenamiento en el sistema, así mismo, al participar el usuario en la prueba se puede verificar y asegurar que el sistema cumplía con los requisitos funcionales.

**Implantación:** No es más que el proceso de establecer y poner en ejecución un sistema determinado para un usuario dado. Un aspecto importante lo constituye el

entrenamiento y formación del personal, ya que pueden necesitar de una formación adicional y un entrenamiento en el nuevo sistema, esto no solo servirá para mejorar el rendimiento del nuevo sistema, sino además para contribuir a la detección de posibles problemas y aumentar la cooperación de los usuarios, reduciendo así las posibilidades de fallo o rechazo al sistema

**Mantenimiento:** Las actividades de mantenimiento se prolongan el resto del ciclo de vida hasta que el sistema sea sustituido por otro.

### 1.1.3.2 Método del análisis estructurado.

El análisis estructurado se concentra en especificar lo que se quiere que haga el sistema o la aplicación, más bien permite que las personas observen elementos lógicos (lo que hará el sistema) separados de los elementos físicos (computadoras, sistemas de almacenamiento etc.). ([Senn, 1990](#)).

#### Elementos del análisis estructurado

Los elementos esenciales son símbolos gráficos, diagrama de flujo de datos y el diccionario de datos.

**Descripción gráfica** se realiza un bosquejo del sistema donde se señalen las características, se identifique las funciones para las que sirve e indique cómo este interactúa con otros elementos. Se utiliza símbolos e íconos para crear un modelo gráfico del sistema.

**Diagrama de flujos de datos** el modelo del sistema recibe el nombre de diagrama de flujo de datos. La descripción de un sistema está formada por un conjunto de diagramas de flujos de datos, es decir se desglosa el modelo del sistema en flujos de datos detallados y fáciles de entender. Ilustra una de las técnicas para representar “Soluciones” a problemas del Mundo Real en forma visual, es decir; en forma grafica. Esta técnica mediante graficas de Diagrama de Flujo, ilustra como diseñar los procedimientos o sentencias con coherencia lógica, que representan la solución al problema planteado.

**Diccionario de datos** todas las definiciones de los elementos en el sistema.

### 1.1.3.3 Método del prototipo de sistemas.

Desarrollo iterativo o en continua evolución donde el usuario participa directamente en el proceso. Los usuarios evalúan el diseño y la información generada por el sistema. ([Senn, 1990](#)).

**Pasos a seguir en el proceso de desarrollo de prototipos son los siguientes:**

- 1- Identificar requerimientos de información que el usuario conoce junto con las características necesarias del sistema.
- 2- Desarrollar un prototipo que funcione.
- 3- Utilizar el prototipo anotando las necesidades de cambios y mejoras. Esto expande la lista de los requerimientos de sistemas conocidos.
- 4- Revisar el prototipo con base en la información obtenida a través de la experiencia del usuario.
- 5- Repetir los pasos anteriores, hasta obtener un sistema satisfactorio.

Estos métodos se emplean en organizaciones de todos tipos y tamaños, cada método es efectivo cuando se emplea adecuadamente. Es importante aclarar que no existe ningún método correcto para desarrollar un Sistema de Información, pero si existen diferentes formas para producir el sistema correcto para una aplicación. Algunos métodos tienen más éxitos que otros y esto depende de cuándo se emplean, cómo se aplican y de los participantes en el método de desarrollo. El indicador definitivo de un método de desarrollo en particular es aquel que se refiere a los resultados obtenidos y no precisión teórica del método.

Estos métodos se emplean en organizaciones de todos tipos y tamaños, cada método es efectivo cuando se emplea adecuadamente.

## 1.2 Sistemas de Gestión de Información. Caracterización.

Dentro de los Sistemas de Información se encuentran los Sistemas de Gestión de Información alguna de las tendencias de estos sistemas en la actualidad son la cantidad y diversidad de recursos tecnológicos disponibles en la infraestructura global de información además del continuo crecimiento de la cantidad de personas que acceden a las computadoras y sus redes. Hoy en día hay un incremento del uso de los Sistemas de Gestión de Información debido a que estos influyen en la competencia de las organizaciones, es decir, las empresas tendrán más éxito mientras tengan un personal

capacitado para trabajar en el desarrollo de aplicaciones tecnológicas de Gestión de Información, mediante el uso de esta tecnología pues se obtendrá más calidad, un mejor servicio a los clientes, un mejoramiento de la gestión de trabajo, un aumento de la productividad, optimización de tiempo además de competitividad.

*(Ponjuan, 2004) plantea que " La gestión es un proceso, por lo que se definiría la gestión de información como el proceso mediante el cual se obtienen, despliegan o utilizan recursos básicos (económicos, físicos, humanos, materiales) para manejar información dentro y para la sociedad a la que sirve. Tiene como elemento básico la gestión del ciclo de vida de este recurso y ocurre en cualquier organización. Es propia también de unidades especializadas que manejan este recurso en forma intensiva, llamadas unidades de información. El proceso de gestión de información debe ser valorado sistémicamente en diferentes dimensiones y el dominio de sus esencias permite su aplicación en cualquier organización. "*

Partiendo de este concepto de Gestión de la Información y además teniendo en cuenta lo que se analizó anteriormente en cuanto a los Sistemas de Información se llega a la conclusión:

Los Sistemas de Gestión de Información son un caso particular de los Sistemas de Información, una aplicación informática, con las características de un Sistema de Información pero con la diferencia que se gestionan tipos específicos de contenidos de información, o sea, se buscan los aspectos comunes y fundamentales que puedan orientar a los usuarios en cuanto al tema a tratar para resolver algún problema en particular, para ellos se procesa, almacena y publica la información necesaria y con la calidad requerida al alcance de todo el que la solicite, para ayudar en el desarrollo y buen funcionamiento de la empresa o del negocio.

En la construcción de un Sistema de Gestión de Información se combina la tecnología con las empresas trayendo consigo un mejor trabajo en un menor tiempo. Hay que tener en cuenta que su interfaz debe ser amigable para hacerle el trabajo mucho más fácil a los usuarios finales debido a que muchos no presentan prácticamente ningún conocimiento informático.

Estos Sistemas de Información son visualizados y utilizados en muchos niveles por ejemplo en la administración, contabilidad financiera, en la medicina etc. por lo que hay

que tener la total certeza de que se trabaja con la información real. Es válido destacar que en estos sistemas las personas representan un papel fundamental porque diseñan, emplean la tecnología y se comunican con el ambiente (proveedores, usuarios, factores culturales, sociales, tecnológicos, políticos, económicos, personal organizacional de la empresa etc.) e intercambian con él.

### **1.3 Gestión de Proyectos.**

#### **1.3.1 Definiciones.**

([Barbero, 2007](#)) plantea que *“Realizar un proyecto significa cambiar. Quien consigue gestionar adecuadamente sus proyectos, tiene más posibilidades de éxito y, por lo tanto, más probabilidades de sobrevivir. Por eso hoy, las organizaciones no buscan personas capaces de cambiar, de adaptarse a los cambios. Buscan personas capaces de realizar los cambios, de liderarlos, de dirigirlos. Personas capaces de llevar a la organización a su siguiente estado en la evolución empresarial. Y por eso, se buscan profesionales en la Gestión de Proyectos.”*

Desde el punto de vista de ([Fernández, 2009](#)) expone que *“La Gestión de Proyectos de Software adquiere mayor importancia aún en el aumento de la complejidad de un proyecto de software donde se quiera lograr como resultado un producto de software de alto nivel de complejidad. En este caso las restricciones de calidad y la falta de planificación en cada una de las etapas de desarrollo del software provocarán el aumento de los costos, el tiempo de desarrollo y el aumento de los riesgos, lo cual pone en peligro el desarrollo del proyecto.”*

El ([Project Management Institute, 2004](#)) se refiere a la Gestión de Proyectos como: *“la aplicación de conocimiento, habilidades, herramientas y técnicas a las actividades de un proyecto para satisfacer los requisitos del proyecto. Se aplican e integran los procesos de dirección de proyectos de inicio, planificación, ejecución, seguimiento y control y cierre. En este proceso el director del proyecto es la persona responsable de alcanzar los objetivos del proyecto”.*

La Gestión de Proyectos tiene como finalidad principal la planificación, el seguimiento y control de las actividades y de los recursos humanos y materiales que intervienen en el desarrollo de un Sistema de Información. Como consecuencia de este control es posible

conocer en todo momento qué problemas se producen y resolverlos o paliarlos de manera inmediata. ([Ajenjo, 2005](#)).

### **1.3.2 Gestión de Proyectos. Antecedentes**

El hombre, desde el principio de su existencia humana, ha realizado proyectos. Y, a través de la historia, se puede ver cómo ha ido aprendiendo, poco a poco, a hacerlo mejor.

Un ejemplo de esto cuando el primer homo sapiens aprendió a encender fuego. Después de más de 300.000 años acarreando brasas encendidas de un campamento a otro, hasta que a alguno de ellos se le ocurrió ponerse a frotar dos palitos y con esto pues la aparición del primer mechero. Evidentemente, eso no es un proyecto, es un descubrimiento, pero de cierta manera esto se relaciona con la Gestión de Proyectos porque fueron documentando cada uno de sus pequeños proyectos, documentaron cómo hacer fuego para enseñárselo a sus hijos, documentaron cómo fabricar una rueda, un hacha. De esta forma, mediante canciones, refranes y burdos dibujos en cuevas, el conocimiento perduró. Cada generación aprendió de las anteriores, y fue mejorando, mejorando y mejorando. ([Barbero, 2007](#)).

Otro ejemplo fue a finales del siglo XIX y principio del siglo XX Henry Gantt ideó la que hasta ahora es la herramienta más característica de la Gestión de Proyectos, el diagrama de Gantt. Inicialmente concebida para gestionar la construcción naval durante la primera guerra mundial, el diagrama de Gantt, permite calcular la fecha de finalización de un proyecto en función de la duración de cada tarea. Son los famosos gráficos de barras. ([Barbero, 2007](#)).

### **1.3.3 Gestión de Proyectos. Características.**

Hoy día se puede ver la aplicación de las técnicas de Gestión de Proyectos en casi todas las áreas de las empresas, finanzas, recursos humanos, producción etc., para que estas empresas logren sobrevivir en el mercado es de vital importancia el éxito de sus proyectos, debido a esto pues hay que tener en cuenta la planificación y la gestión de personas.

A través de la Gestión de Proyectos se puede lograr conseguir más con menos coste, ésta identifica todas las responsabilidades funcionales ante el cumplimiento de la misión de la empresa, asegurándose que todos los miembros de la organización

conozcan su responsabilidad. Así mismo, identifica las posibles mejoras en los procesos, proporcionando ahorros en tiempos y costes. Coordina los diferentes recursos internos y externos. Aporta una visión de conjunto y mejora la comunicación en la empresa. Permite transferir conocimientos entre departamentos que, de otra forma, actuarían de forma estancada. Maneja presupuestos generales y además marca prioridades dentro de las distintas acciones pendientes. Aporta una correcta percepción sobre la auténtica capacidad del equipo, ya que maximiza las sinergias entre los distintos miembros, además es capaz de identificar los riesgos y problemas en fases tempranas, permitiendo que se diseñen acciones correctivas a tiempo.

Cuando se habla de Gestión de Proyectos es bueno destacar que se centra en tres partes Personal, Problema y Proceso.

**Personal:** El factor humano es importante en la investigación de software. Es fundamental tener la gestión del personal con el fin de aumentar la preparación en la organización del software, ayudando a atraer, motivar, y retener al talento necesario para mejorar su capacidad de desarrollo de software. Toda organización que alcanza su madurez en el área de gestión de personal tiene una mayor probabilidad de implementar unas eficaces prácticas de ingeniería de software, esto guía a que las organizaciones tengan un proceso de software maduro.

**Problema:** Se establecen objetivos y se deben considerar soluciones alternativas e identificar las dificultades técnicas de gestión. Con esta información es posible definir unas estimaciones razonables del costo, una subdivisión realista de las tareas del proyecto o una planificación del proyecto asequible que proporcione una indicación fiable del proyecto.

**Proceso:** En el proceso de software se proporciona la estructura desde la que se puede establecer un detallado plan para el desarrollo del software.

La producción de software se convierte en una actividad con más y más demanda, lo que provoca que las empresas o instituciones dedicadas a la producción de bienes y servicios informáticos tiendan a buscar las vías más idóneas para el perfeccionamiento de la gestión empresarial donde la gestión de la información ejerce un rol importante junto a la Gestión de Proyectos. Es imprescindible para los proyectos de software ser

gestionados ya que la construcción de software de computadora es una empresa compleja, particularmente si participa mucha gente, trabajando durante un período de tiempo relativamente largo.

La Gestión de Proyectos no es más que la dinámica del ser humano en función del beneficio común todo con el fin de optimizar recursos y tiempo de trabajo, es imprescindible tener en cuenta la buena selección de los integrantes de un grupo de trabajo y seleccionar correctamente las técnicas de trabajo según los objetivos previstos en el trabajo con el grupo.

Las características del mercado actual, en constante evolución y con un alto grado de incertidumbre, exigen a las empresas una flexibilidad y una capacidad de adaptación a las demandas muy importantes. Las soluciones tecnológicamente avanzadas ofrecen una respuesta rápida, eficaz y de bajo coste. En este entorno de alta competitividad e incertidumbre, la Gestión de Proyectos se convierte en una herramienta de alto valor que permite dirigir con éxito cualquier proceso.

Aspectos más importantes en la Gestión de Proyectos:

Definición de los objetivos del proyecto: en muchos casos, los objetivos que hay que obtener no son suficientemente definidos o no son convenientemente comunicados y por otro lado la elección del equipo de proyecto y su coordinación a lo largo de la vida de éste. Se debe entender las motivaciones de los miembros del equipo, alinearlas con la consecución de los objetivos globales y asegurarnos de que no existen incompatibilidades entre unos y otros.

La Gestión de Proyectos de Software es una actividad protectora dentro de la ingeniería de software. Comienza antes de iniciar cualquier actividad técnica y continua a lo largo de la definición, del desarrollo y del mantenimiento del software. La gestión implica supervisión y control del personal, del proceso y de los eventos que ocurren mientras evoluciona el software desde la fase preliminar a la implementación. ([Pressman, 2005](#)).

### **1.3.4 Sistemas de Gestión de Proyecto.**

Los Sistemas de Gestión de Proyectos son fundamentales en toda empresa que desee el éxito de sus negocios. Estos son capaces realizar un seguimiento activo de los



proyectos, identificar las responsabilidades en el proceso de gestión. A través de estos logramos planear, controlar y mejorar elementos de una organización que influyan en la satisfacción del cliente.

Un Sistema de Gestión de Proyecto ayuda a lograr los objetivos de la organización mediante una serie de estrategias, que incluyen la optimización de procesos, el enfoque centrado en la gestión y el pensamiento disciplinado. Estos sistemas son necesarios porque las empresas de hoy día se enfrentan a muchos retos como por ejemplo la rentabilidad, la competitividad, el crecimiento, la tecnología etc., equilibrar estos y otros requisitos empresariales es una ardua tarea y es donde juegan su papel fundamental los Sistemas de Gestión de Proyectos al permitir aprovechar y desarrollar el potencial existente en la organización.

Con la existencia de un eficiente Sistema de Gestión de Proyectos podría ayudar a gestionar riesgos sociales, ambientales y financieros, mejorar en la efectividad operativa, reducir costos, aumentar satisfacción de clientes y partes interesadas, proteger la marca y la reputación de dicha empresa o entidad, lograr mejoras continuas, potenciar la innovación además de aportar claridad y confianza al mercado y sus seguidores.

## **Capítulo 2. Elección de la herramienta para el desarrollo de la gestión de proyectos en la UCI.**

### **2.1 Estudio comparativo de las herramientas para el desarrollo de Sistemas de Gestión de Proyectos de Software.**

Anteriormente han sido abordadas algunas cuestiones que reflejan la importancia de contar con un Sistema de Gestión que esté soportado por una herramienta que permita llevar a cabo el proceso de desarrollo del software de manera efectiva en la universidad, siendo una proyección para la misma en su afán de organizar el modelo de producción. Para ello a continuación será abordada la esencia de las herramientas de desarrollo colaborativo.

### **2.2 Herramientas de desarrollo colaborativo.**

El desarrollo de software empezó a tomar preponderancia y la programación basada en objetos, también comenzaron a ser comunes las herramientas de control de versiones de código, elementos básicos como el correo electrónico y herramientas de mensajería instantánea en todo grupo de desarrollo de software o software factories.

Este conjunto de componentes actualmente resultan indispensables y los que entre otros, forman el entorno o herramienta de desarrollo colaborativo. Estos entornos atienden las necesidades de sus integrantes como grupo de desarrollo, que pueden o no estar ubicado físicamente en la misma área, departamento, edificio, ciudad o país, favoreciendo a los grupos de trabajo globales.

Estas herramientas son los medios que permiten acceder a ciertos servicios que facilitan a los usuarios comunicarse y trabajar en conjunto sin importar que no estén reunidos en un mismo lugar. Normalmente, con ellos se puede compartir información en determinados formatos (audio, texto, video), y en algunos casos producir conjuntamente nuevos materiales productos de la colaboración. Muchos de ellos proveen de avanzadas funcionalidades que facilitan tareas como publicación de información, búsquedas, accesos, entre otros.

Por lo general las herramientas colaborativas tienen un funcionamiento similar, trabajan bajo el concepto de cliente servidor, donde los clientes son los usuarios que hacen uso de un servicio y el servidor se encarga de ofrecer este servicio. Es por esto que estas herramientas son de gran importancia ya que tienen excelencia para el trabajo a distancia. Herramientas como el email y la mensajería instantánea se volvieron de uso cotidiano estas se desarrollan para usar la Web como plataforma, es decir, no hace falta bajar ni instalar nada. ([Zapata & Julielsy, 2008](#)).

El trabajo colaborativo se define como procesos intencionales de un grupo para alcanzar objetivos específicos, más herramientas diseñadas para dar soporte y facilitar el trabajo las cuales estas a su vez brindan de sus funcionalidades y actividades para que el trabajo colaborativo fluya. ([Crow, 2006](#)).

En el marco de una organización, el trabajo en grupo con soporte tecnológico se presenta como un conjunto de estrategias tendientes a maximizar los resultados y minimizar la pérdida de tiempo e información en beneficio de los objetivos organizacionales.

El mayor desafío, es lograr la motivación y participación activa del recurso humano. Además, deben tenerse en cuenta los aspectos tecnológicos, económico y las políticas de la organización. Trabajo colaborativo palabras para designar el entorno en el cual todos los participantes del proyecto trabajan, colaboran y se ayudan para la realización del proyecto y las herramientas son la base de dicho trabajo.

A pesar de las tantas investigaciones que se han realizado con el objetivo de seleccionar una herramienta para la Gestión de Proyectos, aún no se ha logrado nada definitivo en este tema. Muchos autores coinciden en la importancia de elegir la herramienta que mejor se adecuó a las necesidades de la organización.

Se recomienda utilizar la herramienta que responda como mínimo al 80% de las exigencias de la organización.

### **2.3 Proceso de elección de una herramienta.**

Han sido muchas las investigaciones enfocadas en el problema de seleccionar una herramienta para la Gestión de Proyectos, sin que haya aun nada definitivo en este

tema. Pero en lo que sí todos coinciden es en la importancia de elegir la herramienta que mejor se adecúe a las necesidades de la organización. Se recomienda utilizar la herramienta que responda como mínimo al 80% de sus exigencias. Si no encuentra ninguna lo suficientemente adecuada, se puede optar por crear una propia.

A continuación se presentan algunas de las preguntas que debe plantearse un especialista a la hora de escoger “la mejor herramienta”:

- 1-¿Cómo es el proceso de gestión que se desea establecer?
- 2-¿Cuál es el volumen de datos que se prevé gestionar a corto y mediano plazo?
- 3-¿Cuántas personas van a interactuar y de qué manera?
- 4-¿Dónde se encuentran esas personas?
- 5-¿Es posible utilizar la red de área local como medio de difusión o debo utilizar Internet?
- 6-¿Si fuera necesario modificar la herramienta, qué impacto tendría en cuanto a tiempo y/o complejidad?
- 7-¿Es necesario proteger el acceso al sistema?; ¿pueden todas las personas que tienen acceso hacer lo que quieran?
- 8-¿Qué elementos de información terminológica se pretenden incluir? ¿Es posible incluirlos?
- 9-¿Es tan genérica y/o configurable que soporta el proceso de mi organización?
- 10-¿Cómo será extraída la información de la organización y/o proyecto? ¿Por término? ¿Por sinónimo? ¿Por fecha? ¿Por campo temático?
- 11- ¿Qué campos deben tener los reportes? ¿Permite modificarlos?
- 12-¿En qué idioma(s) serán gestionados los datos?
- 13-¿Qué ocurrirá con los datos que ya existen en otros formatos?
- 14-¿En caso de ser necesario, será posible integrar la herramienta de gestión, con otros sistemas? ¿Qué tan complejo resulta?
- 15-¿Se tendrá que intercambiar información con otras personas?
- 16-¿Qué tipo de sistema computacional (soporte físico y programas) se tiene?
- 17-¿Cuál es el tiempo de respuesta requerido?
- 18-¿Se tendrá que gestionar recursos? ¿Qué tan bien se puede hacer? ¿Se puede definir una manera de hacerlo?

19-¿Cuántas herramientas es posible adquirir para aplicar el proceso de gestión de proyectos de la organización?

20-¿De qué recursos financieros se dispone para comprar o desarrollar una herramienta de gestión?

Conocer el proceso de gestión que se desea seguir, está aun sin respuesta debido a que ahora es que está siendo definido el proceso de gestión de proyectos de la UCI.

Esto conlleva a que los jefes de proyecto no tengan una normativa, lineamiento, política o procedimiento por el cual guiarse a la hora de gestionar sus proyectos, y simplemente adaptan la Gestión del Proyecto a las funcionalidades que les brinda cualquier herramienta de gestión, sin siquiera analizar si se ajusta a sus necesidades.

Elementos a tener en cuenta para la selección de una herramienta.

### **a) Establecer la necesidad:**

Hacer un estricto diagnóstico de lo que la organización necesita y dejar que esto sea lo que genere cualquier decisión acerca de la utilidad de alguna herramienta.

En la UCI, específicamente estas necesidades fueron identificadas con apoyo principalmente del proceso de mejora, la guía del PMBOK y en menor medida en los materiales de la Maestría de Gestión de Proyectos de la Universidad. Llegar a la base, proyectó, permitió identificar gran parte de los problemas; las entrevistas y talleres con jefes de proyectos y la participación en auditorías realizadas fueron de gran importancia para detectar las problemáticas. ([Dirección Técnica.doc, 2007](#)).

### **b) No guiarse por la última moda:**

Las herramientas deben estar bien alineadas con los objetivos corporativos. Muchas herramientas usadas incorrectamente, no sólo no tienen éxito, sino también incrementan considerablemente el escepticismo de los empleados acerca de iniciativas en el futuro. Como se explicaba anteriormente, la carencia de un proceso definido de Gestión de Proyectos para la UCI, ha dado paso a la innovación o simplemente ha permitido la aplicación de la ley del menor esfuerzo realizando solamente algunas

actividades de la Gestión de Proyecto, que les permite la herramienta que usan. Estas herramientas por lo general son utilizadas incorrecta o parcialmente, y por lo general el único criterio de selección a que fue sometida, fue la sugerencia de un conocido o “la fama” que tiene dicha herramienta en determinada comunidad.

### **c) Obtener los hechos:**

Asegurarse de conocer la herramienta y los riesgos que ella conlleva. Este aspecto es muy importante, puesto que la mala utilización de alguna herramienta puede conllevar incluso, a su sustitución sin otro criterio que la simplicidad; hay veces en que es preferible utilizar una herramienta que soporte en un grado considerable el proceso de la organización, y que no sea tan intuitiva como otra, que a pesar de estar muy insipiente es agradable al usuario y fácil de utilizar.

### **d) La adaptación generalmente mejora la herramienta:**

Las herramientas estándar rara vez no toman en cuenta las particularidades del mercado o de la organización. Las herramientas son el medio para lograr una meta, no un fin por sí mismas, por tanto, el proceso de adaptación de una herramienta ve su fase de mejora a la hora de seleccionar una ya creada y reutilizarla teniendo en cuenta las propias exigencias y necesidades de la organización donde se vaya a implementar.

La UCI particularmente por ser una universidad atípica por sus propias características se ve en la necesidad en la mayoría de los casos de crear comunidades internas capaces de adaptar cualquier herramienta a las exigencias de su entorno.

## **2.4 Herramientas de Gestión de Proyecto utilizadas en la UCI.**

En la UCI se han realizado esfuerzos encaminados a la obtención de una herramienta de Gestión de Proyectos, sobre todo más ajustada a sus necesidades, entre estos casos se encuentran:

CENTALAB. Adaptación del Redmine e integración con el generador de reportes.

FACULTAD 10. Proyecto de Adaptación de DotProject: Desarrollo de módulos importantes para DotProject.

FACULTAD 3. POLO de Gestión Gubernamental. Proyecto SIGESPRO: cuyo objetivo es obtener un producto tipo que permita dar soluciones de forma ágil a diferentes clientes en la Gestión de sus Proyectos y cumpla además con los lineamientos de producción de software definidos en la universidad.

Como se explicaba anteriormente en la UCI se han utilizado un aproximado de 7 herramientas diferentes como se muestra en la siguiente tabla:

Herramienta de Gestión de Proyecto	Cantidad de Proyectos
Dot Project	50
Redmine	12
Gforge	11
Project.Net	11
Trac	9
Gantt Project	2
OpenProj	1

**Table 1: Diagnóstico realizado por los especialistas de la Dirección Técnica.**

A continuación se realizará un estudio comparativo para conocer un poco más acerca de dichas herramientas.

**GForge:** Es una referencia en lo que a desarrollo colaborativo se refiere. Es un ambiente rico y maduro. Los servicios están condicionados por herramientas agregadas centralmente y presentados en un esquema con un diseño simple y estático. Utiliza un mecanismo de plug-in, lo que brinda facilidades para el desarrollo de funcionalidades independientes, aunque al mismo tiempo exige conocimiento de los administradores de la plataforma en cuanto a los plug-ins que sean útiles a sus propósitos.

Nombre	GForge
URL	<a href="http://gforge.org/">http://gforge.org/</a>
Licencia	GPL
Última versión estable	5.5 (2008-09)

**Table 2 Descripción del GForge.**

### Funcionalidades

Las prestaciones fundamentales del GForge son:

- 'My Page' muestra todas sus tareas, rastreador de elementos, foros y paquetes de archivos monitoreados a través de todos los proyectos en el sistema.
- Número ilimitado de proyectos, cada uno con su propio SCV o repositorio de Subversion automáticamente creado.
- Acceso al control de los repositorios.
- Estadísticas obtenidas de los repositorios.
- Listas de correo.
- Foros (posibilidad a postear vía email), con soporte a varios modos de vista (El nested, flat, threaded, y ultimate).
- Rastreadores (posibilidad de responder a los elementos rastreados por email).
- Acceso basado en el control de roles que permite una sencilla configuración de permisos para los miembros de los proyectos.
- Administrador de tareas.
- Gráficas de Gantt.
- Administración de documentos con cola de aprobación.
- Encuestas.
- Noticias.
- Sistema de liberación de archivos.



- Wiki.

**Redmine:** es una aplicación Web multiplataforma para la Gestión de Proyectos, liberado bajo la licencia GNU v2, y que ofrece facilidades de instalación y uso. Sin embargo en la universidad existe poca experiencia en cuanto a su uso, solo ha sido probada por miembros del grupo de Arquitectura de la Dirección Técnica.

Nombre	Redmine
<b>URL</b>	<a href="http://www.redmine.org">http://www.redmine.org</a>
<b>Licencia</b>	GPL(v2)
<b>Última versión estable</b>	0.8.3

**Table 3 Descripción del Redmine.**

A continuación se listan algunas de sus características más relevantes:

- Fácil instalación y configuración.
- Soporte multiproyectos.
- Control de acceso basado en roles.
- Administrador de tareas flexible.
- Diagrama de Gantt, Calendario.
- Noticias, documentos y gestor de archivos.
- Notificaciones por correo electrónico y canales RSS.
- Wiki para cada proyecto.
- Foro para cada proyecto.
- Seguimiento de la planificación del tiempo
- Permite personalización del tiempo, tareas, usuarios y proyectos
- Sistema de Control de Versiones (SVN, CVS, Git, Mercurial, Bazaar and Darcs)
- Creación de tareas por email

- Soporte para autenticación con LDAP.
- Soporta múltiples bases de datos.

Wiki

Cronología

Mapa de rutas

Navegador de repositorio de código

Rastreador de errores

Búsqueda

Ofrece un ambiente centralizado, los servicios están concentrados en un servidor y están orientados a la comunidad, tienen “un proyecto, una comunidad y un servidor”, con la intención de fomentar la apropiación por parte de la comunidad. Una desventaja fundamental es que no ofrece soporte multiproyectos en la actualidad, aunque se proyecta en ese sentido. Representa un proyecto en crecimiento, pero que todavía no posee un estado de madurez avanzado.

**DotProject:** fue creado por dotmarketing.org en el año 2000, con el fin de construir una herramienta para la Gestión de Proyectos. “DotProject” esta construido por aplicaciones de Código abierto y es mantenida por un pequeño pero dedicado grupo de voluntarios. Es una aplicación basada en web, multiusuario, soporta varios lenguajes de software Esta programada en PHP, y utiliza MySQL como base de datos (aunque otros motores como Postgres también pueden ser utilizados). La plataforma recomendada para utilizar dotProject se denomina LAMP (Linux + Apache + MySQL + PHP). De todas formas,

existen binarios para instalar DotProject bajo otros sistemas operativos tales como Microsoft Windows (NT, 2000, XP) y Mac.

Nombre	Dot Project
URL	<a href="http://Dotroyect.edgwall.org/">http://Dotroyect.edgwall.org/</a>
Licencia	Código Abierto(GPL)
Última versión estable	0.11.5

**Table 4 Descripción del DotProject.**

### Funcionalidades

El grupo que desarrolla DotProject basa su espíritu de trabajo en los siguientes puntos:

- Proveer a los usuarios de funcionalidad orientada a la Gestión de Proyectos.
- Construir una herramienta con una interfaz de usuario simple, clara y consistente.
- Ser de código abierto, libre acceso y utilización.

El software es libre para todo quien quiera descargarlo. Además, existe soporte gratuito provisto por voluntarios en los foros de la comunidad y vía la documentación online. También es posible contratar soporte del equipo de desarrollo a través del llamado "Foro de Soporte Prioritario".

Como ya se ha mencionado anteriormente, DotProject es una herramienta orientada a la Gestión de Proyectos. Para eso se orienta a la administración de recursos para

desarrollar un producto, cuya producción requiera de un conjunto de actividades o tareas que se desarrollen entre ellas en forma paralela o independiente.

La aplicación consta de un conjunto de entidades ordenadas jerárquicamente las cuales permiten brindar la funcionalidad del producto.

A continuación se mencionan las entidades más importantes de DotProject:

- Compañías: Son las entidades que agrupan proyectos, actividades y usuarios.
- Departamentos: Son áreas dentro de las compañías, que permiten agrupar usuarios en dicho nivel.
- Usuarios/Contactos: DotProject tiene usuarios los cuales son capaces de loguearse a DotProject y trabajar dentro del esquema de permisos que posea el rol de dicho usuario. Los contactos son usuarios especiales que asignados a un determinado proyecto pueden recibir por ejemplo: correo, actualizaciones y noticias pero no necesariamente deben tener acceso al sistema DotProject. Los usuarios y contactos pertenecen a una compañía.
- Proyectos: Es la entidad que contiene el grupo de tareas necesarias para desarrollar un determinado producto o servicio.
- Actividades: Son las tareas asignadas dentro de un proyecto. Son los componentes sobre los cuales se controla: la duración, dependencias, recursos asignados y progreso. Las actividades deben de pertenecer a un único proyecto.
- Diagramas de Gantt: Permite ver en forma gráfica las actividades ordenadas jerárquicamente, mostrando las dependencias y solapamientos de las mismas.
- Tickets: Para administrar todos los problemas relacionados a un proyecto.
- Archivos: Permite almacenar archivos dentro de un proyecto permitiendo un versionado básico de los mismos.
- Foros: Permite la creación de foros de discusión dentro de cada proyecto para distribuir información y discutir temas relativos al proyecto del foro.

- Administración del sistema: Contiene las actividades relacionadas a la administración de usuarios, roles y configuración del sistema.
- Recursos: Permite asignar recursos no humanos (oficinas, equipamiento, etc.) a un proyecto.

**Project.Net** En la escala de aplicaciones de empresa de Gestión de Proyectos para Microsoft Windows y sistemas operativos \* Unix.

Project.Net fue fundado en 1999 para desarrollar aplicaciones de colaboración de proyecto usando tecnologías de la Internet. El foco inicial de la compañía construiría un motor de colaboración para usar antes de los cambios públicos y privados basados en la Web. En 2002, la Revista la PC recompensó a Project.Net con el premio De Primera Calidad de los Editores en una revisión de aplicaciones de la gerencia de proyecto basado en la Web.

Project.Net esta actualmente en uso por más de 50,000 personas en todo el mundo para ayudar a manejar sus proyectos. La revista Comercial universitaria publicó un artículo en la Dirección de Proyecto y del Portafolio para la que revisa la necesidad y el uso de Project.Net en el departamento de facilidades en Cornell University.

Nombre	Project.Net
URL	<a href="http://project-net.edgewall.org/">http://project-net.edgewall.org/</a>
Licencia	propietaria
Última versión estable	-*-

Table 5 Descripción del Project.Net.

### 2.5 Resultados del estudio comparativo.

Del conjunto de herramientas anteriores se escogieron las 3 que a consideración de los autores se ajustan mejor al proceso definido, estas son: Project.Net, Redmine y DotProject. En lo que basándonos en un estudio realizado por parte de la Dirección Técnica se elaboró una lista de requerimientos extraída del programa de mejora, PMBOK, CMMI for Developers; tratando de abarcar todas las áreas de la Gestión de Proyectos y dar cobertura a cualquier cambio que pueda sufrir el proceso.

La lista de elementos automatizables de Gestión de Proyectos (como se denominó) fue revisada y aprobada como un buen acercamiento, por varios especialistas en el tema como por ejemplo: Dr. Roberto Delgado, profesor consultante de la CUJAE del MES, Consultor de Dirección Integrada por Proyectos y Asesor DCT. MES; Yamila Vigil Regalado, la vicedecana de la MiniUCI de Artemisa; Msc. Maypher Román Durán, Jefe de Departamento de Gestión de Proyectos en la UCI; etc.

Posteriormente se elaboró una tabla con la lista de requerimientos y se les envió a los jefes de proyecto que a entender de los autores son los más capacitados en el uso y modificación de las herramientas.

A partir de la herramienta que usaban cada uno en sus proyectos, llenaron la tabla respondiendo el nivel de cumplimiento de cada requerimiento. Los valores variaban

Desde: NO, NO SÉ, INSUFICIENTE y SUFICIENTE.

**SUFICIENTE:** La herramienta implementa la funcionalidad evaluada de forma nativa, o es muy sencillo ejecutarla con las bondades que presenta (elementos personalizables).

**INSUFICIENTE:** La herramienta implementa la funcionalidad parcialmente de forma nativa, o se puede ejecutar de alguna forma.

**NO:** Significa que no hay forma de ejecutar la funcionalidad con la herramienta.

**NO SÉ:** Usted no sabe si se puede o no ejecutar la funcionalidad evaluada.

	ProjectNet		Redmine		DotProject	
	Cantidad	%	Cantidad	%	Cantidad	%
No	111	52.35	142	66.98	139	65.56
Suficiente	51	24.05	40	18.86	59	27.83
Insuficiente	49	23.11	13	6.13	10	4.71
No sé	1	0.47	17	8.01	4	1.88
Total	212	100	212	100	212	100

**Table 6 Diagnóstico realizado por los especialistas de la Dirección Técnica.**

Esta forma de adquirir la información se utilizó con el doble propósito de primero, obtener información valiosa de cada una de las herramientas y segundo que cada uno se percatara de lo que le faltaba.

A partir de esto se realizó una comparación buscando obtener de las 3 la herramienta que más puede ajustarse al proceso de Gestión de Proyectos de la UCI, en el anexo se muestra la comparación en su totalidad. Obteniéndose la siguiente información:

De la tabla anterior se puede advertir que de las 3 la que más soporta el proceso es el Project.Net con un 47 % obtenido de sumar el por ciento de requerimientos soportados suficientemente con el por ciento de los insuficientes, estos últimos se tomaron en cuenta porque pueden ser satisfechos con un mínimo de modificación de la herramienta o mediante alguna acción creativa en su manipulación. Luego está el DotProject con un 31 % de cumplimiento, obtenido mediante el mismo procesamiento y por último el Redmine con solamente un 24%. Estos valores tan minúsculos se deben a que la lista de requerimientos en cuestión contenía nada menos que 212 aspectos a considerar, pudiera considerarse un número elevado pero aun sigue siendo una aproximación, principalmente se buscaba tener un control de las funcionalidades que se necesitarían incluirle a la herramienta y/o modificarle. Es decir, el por ciento obtenido como resultado, nos brinda información de varias cosas al mismo tiempo, por ejemplo: da una medida de lo ajustada que está la herramienta al proceso de desarrollo.

A raíz de las investigaciones realizadas en la universidad; se escogió como herramientas candidatas el Project.Net como principal candidato y el Redmine como primera variante ya que Dot Project no presenta una actualización de su comunidad desde el año 2007 además el sistema se hace algo confuso en bastantes módulos. Las traducciones al español no están completas, incongruencias al modificar la base de

datos, por lo que generó fallos graves, con bucle infinito incluido. Presenta carencia de plugins, y las versiones existentes no soportan íntegramente el proceso de gestión llevado a cabo actualmente en la UCI.

	Herramienta	
	RedMine	Project.Net
Debilidades	<ul style="list-style-type: none"> <li>• Lenguaje de Programación (Ruby).</li> <li>• Algunas limitaciones para la Gestión de Proyectos.</li> </ul>	<ul style="list-style-type: none"> <li>• Muy poco conocimiento sobre la herramienta en la UCI.</li> <li>• Base de Datos Oracle por lo tanto no tiene integración con el Generador de Reportes de Centalad.</li> </ul>
Amenazas		<ul style="list-style-type: none"> <li>• Tiempo necesario para estudiar la herramienta.</li> <li>• Capacidad actual para resolver cualquier problema que surja en la utilización de la herramienta.</li> <li>• Impacto en los proyectos de la UCI de la utilización de la herramienta.</li> </ul>
Fortalezas	<ul style="list-style-type: none"> <li>• Código Abierto, Licencia GPL.</li> <li>• Muy intuitiva y fácil de utilizar.</li> <li>• Base de Dato en PostgreSQL, SGBD oficial de la UCI que le permite total integración con Generador de Reportes de Centalad.</li> <li>• Se estará pilotando en el programa de mejoras.</li> <li>• Se cuenta con un amplio conocimiento de la herramienta en la UCI, y será posible resolver cualquier problema que surja.</li> </ul>	<ul style="list-style-type: none"> <li>• Lenguaje de Programación (Java).</li> <li>• Potente herramienta de GP.</li> </ul>
Oportunidades	<ul style="list-style-type: none"> <li>• Comenzar Gestionar los proyectos lo más rápido posible, hasta que exista una mejor opción.</li> </ul>	<ul style="list-style-type: none"> <li>• La reciente instalación de la Herramienta en el entorno de prueba de la DT, permitirá estudiarla a fondo para tomar ideas arquitectónicas o una futura implantación de la herramienta en la UCI.</li> <li>• Con la implantación de la Herramienta se garantizara un</li> </ul>



		entorno mas robusto de GP para la UCI. <ul style="list-style-type: none"><li>• Integrabilidad garantizada con otros sistemas debido a las bondades de la plataforma java.</li></ul>
--	--	---

### 2.6 Propuesta de herramienta a utilizar.

La investigación realizada muestra que las herramientas que fueron analizadas, son las que mayores posibilidades tienen entre todas las que se utilizan en la universidad y que incluso estas se quedan muy por debajo de lo que se necesita.

Debido a esto es necesario comenzar la elaboración de una herramienta propia de Gestión de Proyectos, ya sea adaptando una de las anteriores o implementándola desde cero. Es importante añadir además, que debido a que la elaboración de una herramienta de este tipo, requiere un tiempo considerable, se propone utilizar una herramienta propia de Gestión de Proyectos como se menciona anteriormente, con el fin de adaptarla a las necesidades de la Universidad para implantarlas en la misma, la cual será Redmine que desde el punto de vista de los autores es la que mas se adapta a dichas necesidades.

#### 2.6.1 Arquitectura de Redmine.

Redmine esta soportado sobre un framework llamado Rails cual basa su funcionalidad en aplicaciones web de código abierto escrito en el lenguaje de programación Ruby, siguiendo el paradigma de la arquitectura Modelo Vista Controlador (MVC). Trata de combinar la simplicidad con la posibilidad de desarrollar aplicaciones del mundo real escribiendo menos código que con otros frameworks y con un mínimo de configuración.

El lenguaje de programación Ruby permite la meta programación, de la cual Rails hace uso, lo que resulta en una sintaxis que muchos de sus usuarios encuentran muy legible. Rails se distribuye a través de RubyGems, que es el formato oficial de paquete y canal de distribución de bibliotecas y aplicaciones Ruby.

Rails trabaja con varios servidores web y bases de datos. Para el servidor web, recomendamos Apache. Para la base de datos, puedes usar MySQL, PostgreSQL,

SQLite, Oracle, SQL Server. Funcionará sobre cualquier sistema operativo, pero recomendamos alguno basado en Unix para el desarrollo.

### ¿Quién esta Usando Rails?

Todo el mundo desde organizaciones sin ánimo de lucro pasando por compañías están usando Rails. Rails es todo acerca de la infraestructura así que se acopla perfectamente en cualquier tipo de aplicación Web Es software para colaboración, comunidad, e-comercio, CMS, estadísticas y control.

#### 2.6.2 Arquitectura Modelo-Vista-Controlador (MVC) de Rails.

En el siguiente epígrafe se abordara todo lo referido a la arquitectura del framework que soporta al Redmine. Esto brindará grandes beneficios a venideros desarrolladores, los cuales permitirán que cuando se enfrenten por primera vez a un trabajo con el mismo, no desvíen su atención en un sentido erróneo por falta de conocimiento. A continuación, se explica cada parte de dicha arquitectura.

#### Modelos

Referidos a los datos y generalmente representados por una tabla en la base de datos. En Rails, se crea una tabla para cada modelo, y éste es el responsable de definir las relaciones con otros modelos. Así por ejemplo, el Libro, tendrá una Categoría. Ambos son modelos y estarán representados por tablas en la base de datos.

¿Qué hacen los modelos aparte de "existir"?

Estos modelos presentan unas series de funcionalidades que los hacen únicos dentro de esta relación, a continuación se expresa detalladamente dichas actividades.

**Relaciones:** El modelo es quien declara su relación con otros modelos: por ejemplo, Libro pertenece a varias categorías (belongs to) y Categoría tiene varios libros (has\_many).

**Validaciones:** Los modelos, también se encargan de indicar qué datos son necesarios para su funcionamiento. Por ejemplo, el modelo Libro, compuesto de Título, Páginas y Precio puede declarar que no puede existir sin Título (validates\_presence\_of: title) y que

el precio debe ser número (`validates_numericality_of: precio`) respecto a que sea mayor de 1 se trata en otra parte, dentro del modelo).

### **Controladores**

Son los directores de orquesta. Una vez que les llega una petición desde un navegador, reflejada en la URL, se ponen en marcha realizando operaciones de manipulación de los datos descritos en los modelos: Creación, Actualización, Eliminación o Búsqueda, operaciones típicas de cualquier sistema interactivo (Denominado CRUD o Create, Retrieve, Update, Delete).

<http://rubyonrails.org/>

Una vez realizada la operación, el controlador devuelve una respuesta que generalmente se refleja en una "Vista" o página.

### **Vista**

En MVC, Vista es la lógica de visualización, o cómo se muestran los datos de las clases del *Controlador*. Con frecuencia en las aplicaciones web la vista consiste en una cantidad mínima de código incluido en HTML.

Existen en la actualidad muchas maneras de gestionar las vistas. El método que se emplea en Rails por defecto es usar Ruby Embebido (`archivos.rhtml`, desde la versión 2.x en adelante de RoR `archivos.html.erb`), que son básicamente fragmentos de código HTML con algo de código en Ruby, siguiendo una sintaxis similar a JSP. También pueden construirse vistas en HTML y XML.

Es necesario escribir un pequeño fragmento de código en HTML para cada método del controlador que necesita mostrar información al usuario. El "maquetado" o distribución de los elementos de la página se describe separadamente de la acción del controlador y los fragmentos pueden invocarse unos a otros.

### Controlador

En MVC, las clases del Controlador responden a la interacción del usuario e invocan a la lógica de la aplicación, que a su vez manipula los datos de las clases del Modelo y muestra los resultados usando las *Vistas*. En las aplicaciones web basadas en MVC, los métodos del controlador son invocados por el usuario usando el navegador web.

La implementación del Controlador es manejada por el ActionPack de Rails, que contiene la clase ApplicationController. Una aplicación Rails simplemente hereda de esta clase y define las acciones necesarias como métodos, que pueden ser invocados desde la web, por lo general en la forma `http://aplicacion/ejemplo/metodo`, que invoca a `EjemploController#metodo`, y presenta los datos usando el archivo de plantilla `/app/views/ejemplo/metodo.html.erb`, a no ser que el método redirija a algún otro lugar.

Rails también proporciona andamiaje, que puede construir rápidamente la mayor parte de la lógica y vistas necesarias para realizar las operaciones más frecuentes sobre su soporte de base de datos.

### Soporte de Bases de Datos

Dada que la arquitectura Rails favorece el uso de bases de datos se recomienda usar un SGBD para almacenamiento de datos. Rails soporta la biblioteca SQLite si no es posible emplear una base de datos. El acceso a la base de datos es totalmente abstracto desde el punto de vista del programador, y Rails gestiona los accesos a la base de datos automáticamente (aunque, si se necesita, se pueden hacer consultas directas en SQL).

Rails intenta mantener la neutralidad con respecto a la base de datos, la portabilidad de la aplicación a diferentes sistemas de base de datos y la reutilización de bases de datos preexistentes. Sin embargo, debido a la diferente naturaleza y prestaciones de los SGBDs el framework no puede garantizar la compatibilidad completa. Se soportan diferentes SGBDs, incluyendo Firebird, MySQL, PostgreSQL, SQLite, IBM DB2, Oracle y Microsoft\_SQL\_Server.

### Entorno de Trabajo

Existen muchas alternativas para trabajar con Ruby on Rails, tanto libres y gratuitas como de pago. A continuación se listan las principales:

- **Aplana:** Multiplataforma. Nació como plug-ins de eclipse para la edición y desarrollo web. Actualmente puedes instalarlo como plug-ins o autónomo de forma independiente. Las últimas versiones están muy bien integradas con Ruby on Rails. A principios de 2010 saldrá la versión 3 con importantes mejoras.
- **Netbeans:** Uno de los más usados, libre y totalmente gratuito. Viene muy bien integrado con JRuby (lo cual es algo lógico pues es un programa de Sun).
- **TextMate:** Sólo para Mac. Es el entorno más usado entre la comunidad Rails. Es de pago pero su potencia y forma de trabajo favorece la producción y desarrollo con Ruby on Rails.
- **Gmate:** Un proyecto libre y gratuito para convertir Gedit -el editor de texto de escritorio Gnome de Linux- en un clon muy aproximado de Textmate. Esto se consigue instalando diferentes plugins, temas y retocando algunas opciones. Al ser gratuito es una opción que está cogiendo muchos adeptos hoy en día.

### 2.7 Descripción de Plug-ins (funcionalidades del Redmine).

La descripción de los plug-ins permite al usuario conocer y familiarizarse mucho más con la herramienta y además le brinda una visión de lo que se tratará en la herramienta así como cuáles son sus objetivos. A continuación se presenta la descripción de cada uno de los plugins que han sido ejecutados hasta el momento en la aplicación.

#### 2.7.1 Reportes.

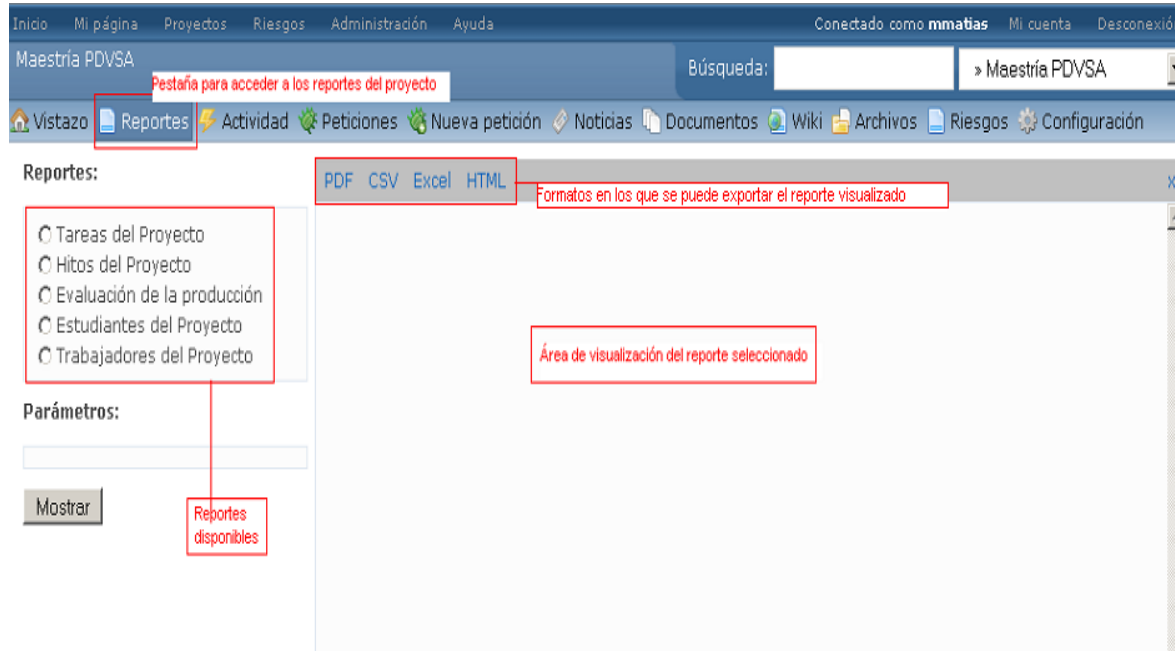
Para acceder a los reportes, solo tiene que ir a la página principal del proyecto (Vistazo), el usuario que lo tenga especificado entre los privilegios de su rol podrá ver el bloque de los reportes, deberá seleccionar el reporte que desee ver, el formato en que lo quiere, el intervalo en que quiere el reporte (por defecto sale la última semana transcurrida, como intervalo) y por último presione el botón mostrar. Paulatinamente se incorporarán reportes y otras funcionalidades al sistema que hoy en vistas de avanzar

en la implantación de la primera versión no están contempladas. Para que un rol tenga acceso a ver los reportes del proyecto o del centro debe tener los permisos necesarios

<input checked="" type="checkbox"/> Añadir notas	<input type="checkbox"/> Modificar notas
<input type="checkbox"/> Modificar notas propias	<input type="checkbox"/> Mover peticiones
<input type="checkbox"/> Borrar peticiones	<input type="checkbox"/> Administrar consultas públicas
<input checked="" type="checkbox"/> Grabar consultas	<input checked="" type="checkbox"/> Ver diagrama de Gantt
<input checked="" type="checkbox"/> Ver calendario	<input type="checkbox"/> Ver lista de seguidores
<input type="checkbox"/> Añadir seguidores	<input type="checkbox"/> Borrar seguidores
<input type="checkbox"/> Administrar relación con otras peticiones	
Noticias	
<input type="checkbox"/> Administrar noticias	<input checked="" type="checkbox"/> Comentar noticias
Reportes	
<input checked="" type="checkbox"/> View reportes	
Repositorio	
<input type="checkbox"/> Administrar repositorio	<input checked="" type="checkbox"/> Hojear repositorio
<input checked="" type="checkbox"/> Ver cambios	<input type="checkbox"/> Acceso de escritura
Risks	
<input type="checkbox"/> Project risks	
Control de tiempo	
<input type="checkbox"/> Anotar tiempo dedicado	<input checked="" type="checkbox"/> Ver tiempo dedicado

**Figure 3 Otorgar permiso de ver reportes.**

Para ver los reportes de un proyecto específico se debe acceder desde el proyecto deseado a la pestaña Reportes



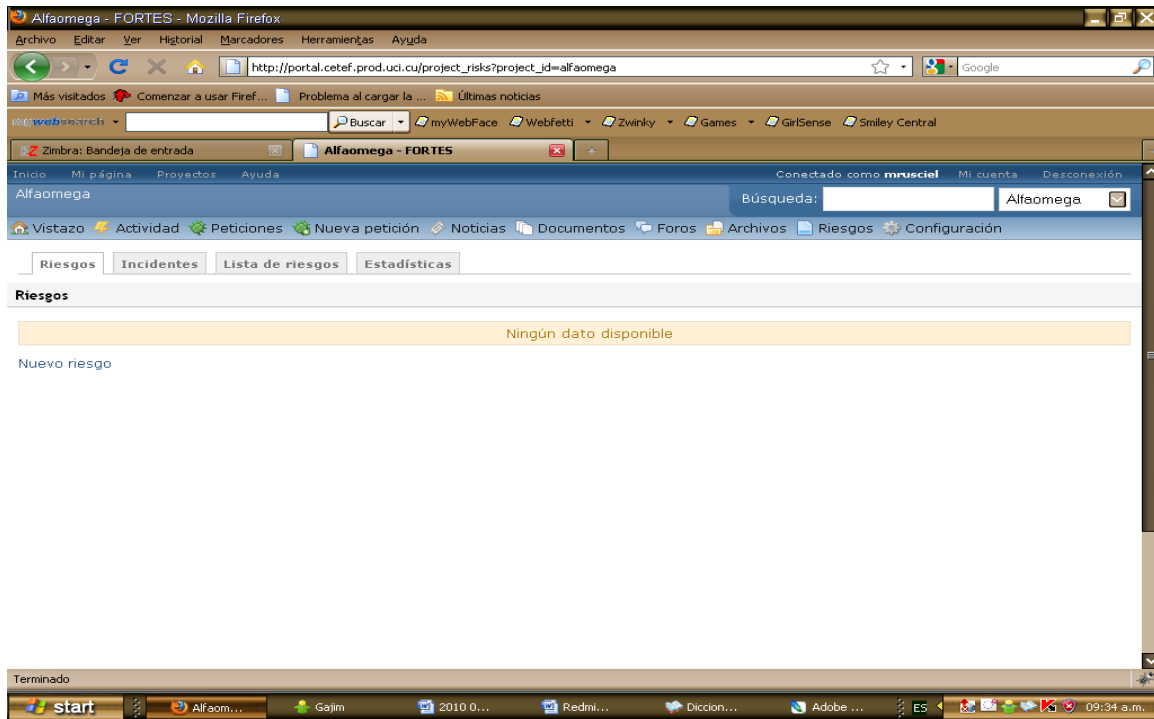
**Figure 4 Visualizar reportes de un proyecto.**

Se debe marcar el reporte que se desea ver entre los que están disponibles y dar clic sobre el botón Mostrar, el reporte saldrá en el área en blanco que se encuentra a la derecha. El reporte visualizado puede ser exportado a diferentes formatos (PDF, CSV, Excel, HTML) y guardarse en la dirección deseada.

## 2.7.2 Riesgos.

1. La persona encargada de la administración del Redmine en el Centro debe:

- a. Los roles que tendrán acceso al módulo de Riesgos.
- b. Las categorías de riesgos: clasificación que suele emplearse para agrupar los riesgos.
- c. Los riesgos conocidos: Macro riesgos.



**Figure 5 Plug-in de Riesgo.**

### 2.7.3 Asistencia

El plug-in para el control de la asistencia está incluido en los Redmine de todos los centros.

Para activarlo debes seleccionar un proyecto específico, luego vas a Configuración y de hay seleccionas la pestaña Módulos, ahí marcas la opción Asistencia.

Luego cuando entres de nuevo a ese proyecto ya tendrás las opciones Asistencia y Revisar asistencia.

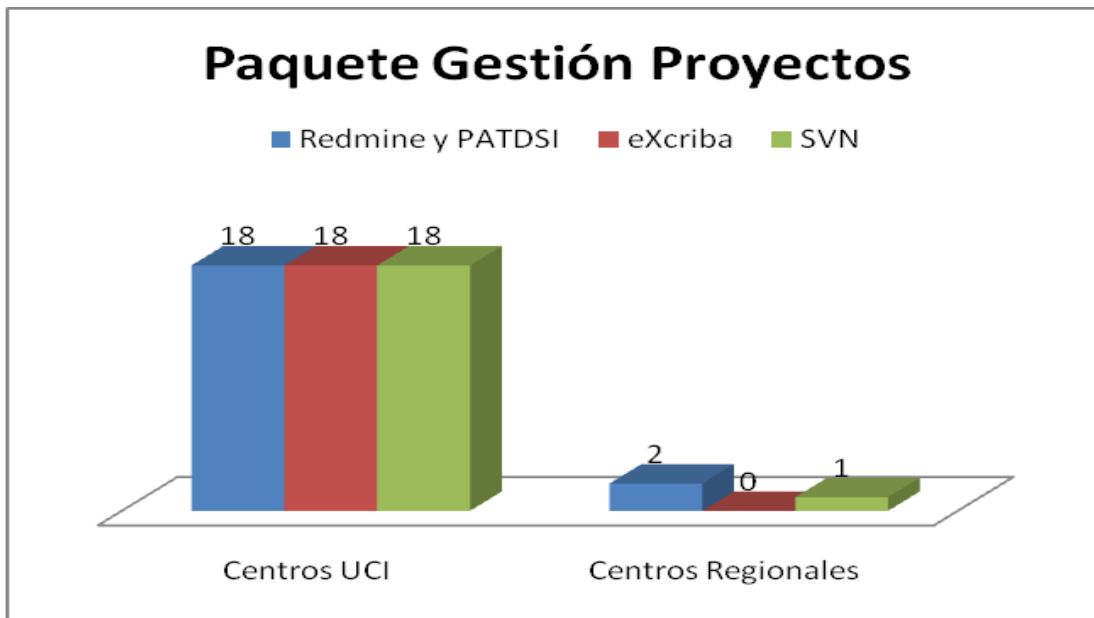
Para pasar la Asistencia debes seleccionar la opción Asistencia si lo haces por primea vez luego debes definir la o las secciones de trabajo y a quienes les vas a pasar asistencia.



## Capítulo 3. Validación de la Herramienta y puesta en práctica.

En este capítulo se representa a través de gráficas de barras la utilización del Redmine y otras herramientas de entornos colaborativos en la UCI donde se evidencia el alto grado de utilización que tiene la aplicación en las diferentes ramas que lo utilizan. También se recogen las direcciones de los diferentes sitios donde ya esta puesto en práctica el software entre otras estadísticas.

<b>Dirección Técnica</b>	<b>Comunidades</b>	<b>CALISOFT</b>
<a href="https://portal.calisoft.prod.uci.cu/">https://portal.calisoft.prod.uci.cu/</a>	<a href="https://comunidades.uci.cu/">https://comunidades.uci.cu/</a>	<a href="https://portal.dt.prod.uci.cu/">https://portal.dt.prod.uci.cu/</a>
<b>CEIGE</b>	<b>CEDAE</b>	<b>CEGEL</b>
<a href="https://portal.cegel.prod.uci.cu/">https://portal.cegel.prod.uci.cu/</a>	<a href="https://portal.cedae.prod.uci.cu/">https://portal.cedae.prod.uci.cu/</a>	<a href="https://portal.ceige.prod.uci.cu/">https://portal.ceige.prod.uci.cu/</a>
<b>GEITEL</b>	<b>GEYSED</b>	
<a href="https://portal.geitel.prod.uci.cu/">https://portal.geitel.prod.uci.cu/</a>	<a href="https://portal.geysed.prod.uci.cu/">https://portal.geysed.prod.uci.cu/</a>	

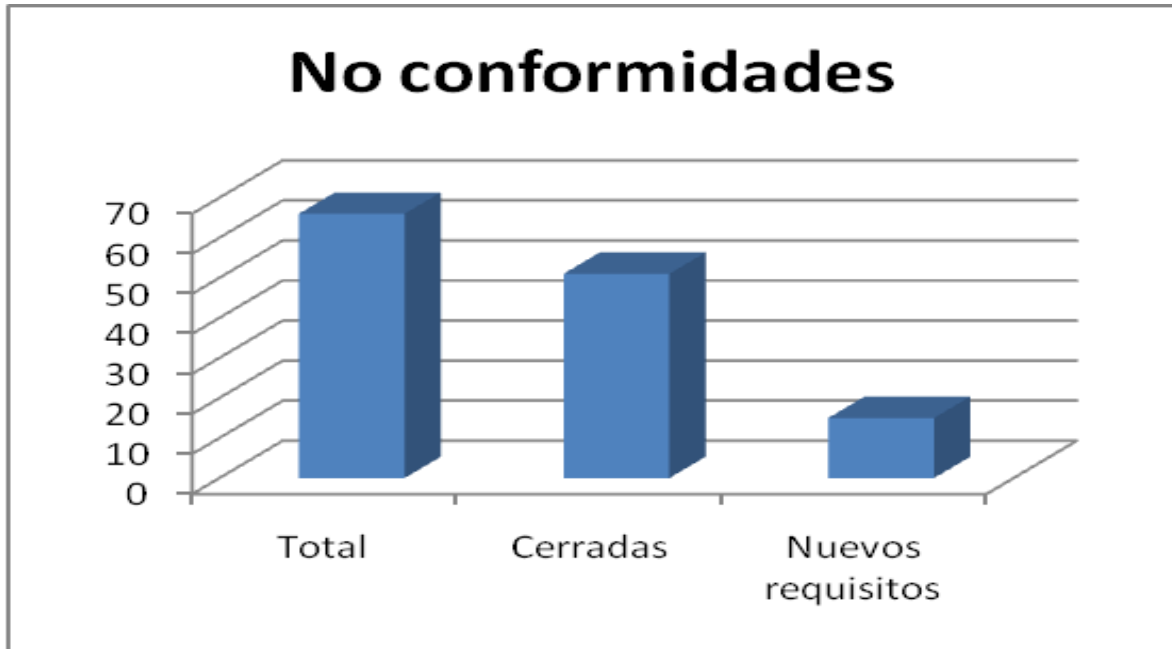


**Table 7 Paquete de Gestión Proyecto.**

El grafico demuestra la gran demanda que posee la herramienta y su nivel de aceptación entre los usuarios.

### 3.1 No conformidades o dificultades que presenta la herramienta.

Estos datos que se muestran a continuación pertenecen a los diferentes usuarios del cual el Redmine se retroalimenta y vale resaltar que ha sido de gran ayuda.



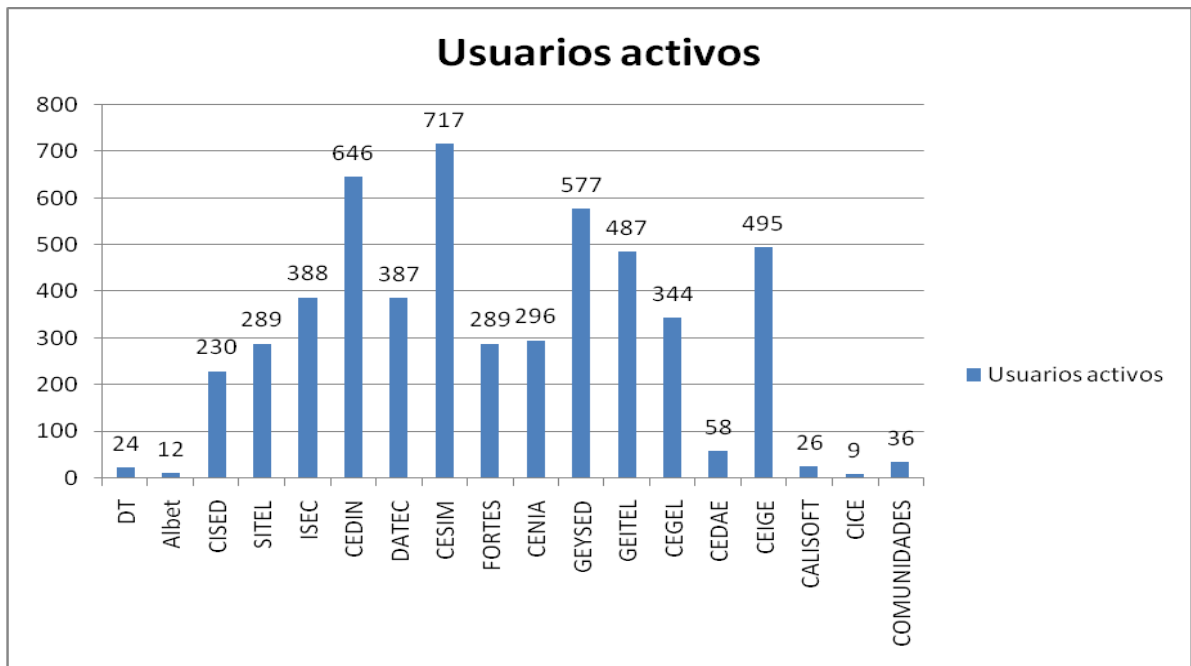
**Table 8 Datos de Centro Soporte abril 2010.**

- Continuar fortaleciendo la capacitación del personal de la UCI en el uso del paquete. (Creación de curso de postgrado).
- Desarrollo del Datamart de Gestión de Proyectos. F6-DATEC.
- Mejora en los reportes existentes y desarrollo de nuevos. Sistema de despacho de necesidades de reportes martes y jueves. Actualización semanal.
- Resolución de nuevos requisitos y no conformidades. (asignar tareas a múltiples usuarios simultáneamente, copiar proyectos, facilidades para la administración). Actualización semanal.
- Integración de nuevos servicios de apoyo a la producción en los portales (servidor de gráficos, Caxtor IDE desarrollo EXT JS, SIGE gestión de costos).

- Integración con la comunidad internacional de desarrollo del Redmine.

### 3.2 Usuarios Activos.

En este epígrafe se recogerá una estadística de la cantidad de usuarios que en la actualidad utilizan la herramienta como se muestra en la siguiente tabla estos usuarios activos se agrupan por Centro de Desarrollo donde la herramienta de entornos colaborativos Redmine se pone en práctica y se muestra al final la cantidad total de activos.



CENTROS	Usuarios activos
DT	24
Albet	12
CISED	230
SITEL	289
ISEC	388
CEDIN	646
DATEC	387
CESIM	717
FORTES	289

CENIA	296
GEYSED	577
GEITEL	487
CEGEL	344
CEDAE	58
CEIGE	495
CALISOFT	26
CICE	9
COMUNIDADES	36
<b>TOTAL</b>	<b>5310</b>

### 3.3 Avances de la herramienta.

En este epígrafe se mostraran los avances que ha tenido el Redmine estadísticamente en un periodo de tiempo el cual se establece en la siguiente gráfica.

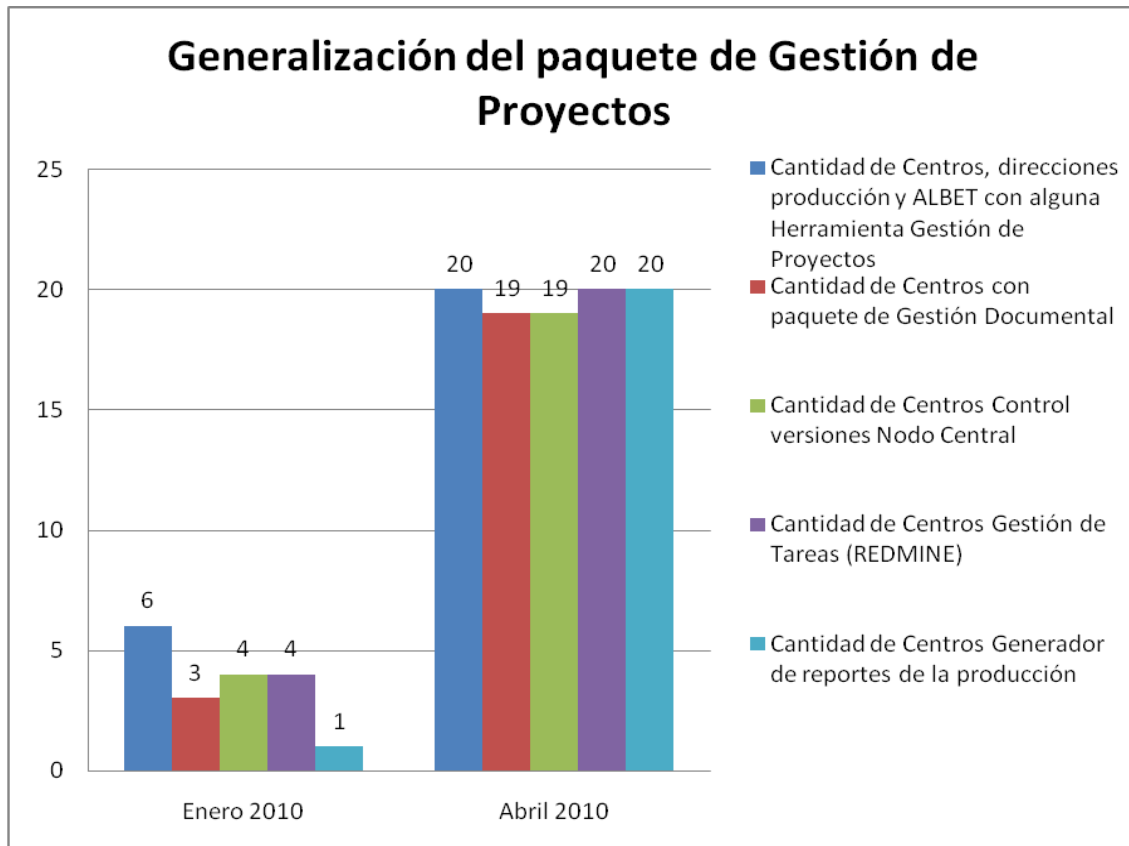
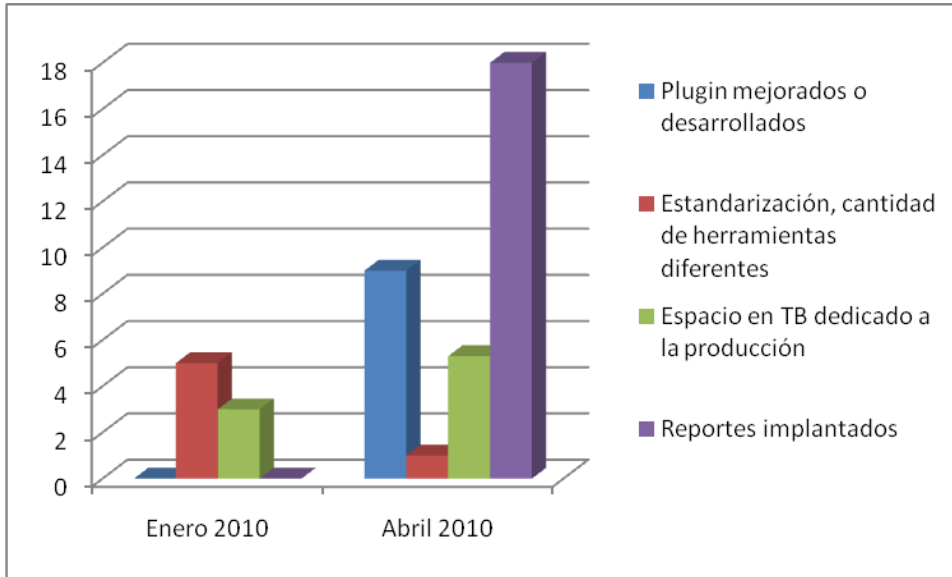


Table 9 Resultado de Avances.



**Table 10 Resultado de Avances.**

Se puede apreciar con datos estadísticamente seguros que esta herramienta es la que más se adecua a lo que hace algún tiempo en la universidad se estaba buscando para poder llevar a cabo la Gestión de Proyectos de Software de la manera más eficiente posible y así poder organizar y optimizar todos los procesos que se gestionan dentro de la universidad aplicando una correcta gestión de software.

### Conclusiones

- Los problemas que más influyen directamente en el fracaso de los proyectos es la mala gestión en los recurso humanos y materiales, la ineficiencia en la planificación de los proyectos, la imperfección de estimaciones en los costos de ejecución y que no se realiza un adecuado seguimiento y control de los riesgos y su mitigación.
- La investigación realizada demostró que existe una enorme variedad de soluciones para la Gestión de Proyectos y que el principal problema es, no encontrar la que mejor se ajusta.
- Se propuso como parte del paquete de herramientas para la Gestión de Proyectos, el Redmine como aplicación de gestión de entornos colaborativos.
- La implantación del Redmine como parte del Paquete de Gestión de Proyectos (GESPRO) demuestra a través de resultados prácticos como está herramienta es capaz de ajustarse a diversos escenarios de la planificación de proyectos.

### **Recomendaciones**

Se recomienda que a partir de esta investigación se continúe profundizando sobre el tema tratado, para adquirir un mayor conocimiento acerca de la Gestión de Proyectos de Software en la universidad.

Continuar con el desarrollando de plug-in para un mejor funcionamiento de la herramienta propuesta.

Modificación de la interfaz para una excelente interacción del usuario con la herramienta.

### Bibliografía Referenciada

1. Ajenjo, A. D. (2005). Dirección y Gestión de Proyectos. Un enfoque práctico (Segunda Edición).
2. Bohem, B. (1984). "Software Engineering Economics". IEEE Transactions on Software Engineering.
3. Börner, K. Chen, C. & Boyack, K. V. (2003). Visualizing knowledge domains. Annual Review of Information Sciences and Technology.
4. Cotec (2002). Empresas y administraciones públicas. El papel de las diferentes herramientas en el fomento de la innovación. Madrid.
5. Dirección Técnica.doc, (2007). Tratamientos y métodos de elección de herramientas.
6. Fernández, Anisleiby. (2009). Análisis y evaluación de la Gestión de Proyectos de Software en entornos académicos – productivos. Universidad de las Ciencias Informáticas.
7. Gil, Manuel-Alejandro. (2007). Proceso de desarrollo de aplicaciones de acuerdo a los conceptos de software libre.
8. Guía de los Fundamentos de la Dirección de Proyectos. (2008). Tercera Edición [Book] / auth. PMI. - [s.l.] : Project Management Institute, Inc., 2004. - Edición de bolsillo – Español.
9. Hernández, R. (2005). Curso Básico de Gestión de Proyectos. La Habana: Universidad de las Ciencias Informáticas.
10. Heredia, R. (1995a). Dirección Integrada de Proyecto-DIP-“Project Management”. Madrid: Escuela Técnica Superior de Ingenieros Industriales.
11. Herramientas. Tutorial de Terminologías [Online] // Herramientas de Gestión de Datos Terminológicos.
12. <http://www.econlink.com.ar/sistemas-informacion/definicion>.
13. <http://patob2000.wordpress.com/2006/09/28/funcionamiento-general-de-las-herramientas-colaborativas/>.
14. <http://www.monografias.com/trabajos7/sisinf/sisinf.shtml#ars>.



## BIBLIOGRAFÍA REFERENCIADA

---

15. [http://www.termiumplus.translationbureau.gc.ca/didacticiel\\_tutorial/espanol/lecon4/page4\\_5\\_8\\_s.html](http://www.termiumplus.translationbureau.gc.ca/didacticiel_tutorial/espanol/lecon4/page4_5_8_s.html).
16. <http://www.econlink.com.ar/sistemas-informacion/definicion>
17. <http://julizv.blogspot.com/2008/02/uso-y-aplicacin-de-las-herramientas.html>. Consultado 19/04/10.
18. Jaque Barbero, Miguel. (2007).Gestión de Proyectos.
19. Ponjuan, Gloria. (2004). Sistemas de Información: Principios y Aplicaciones.
20. Ponjuan, Gloria. (2004).Gestión de información: Dimensiones e implementación para éxito organizacional.
21. Poppendieck, M. (2003). Lean Software Development: An Agile Toolkit for Software Development Managers: Addison Wesley.
22. Pressman (2002). Planificación de Proyectos de Software. In M. Hill (Ed.), Ingeniería del Software. Un enfoque Práctico (Quinta ed.).
23. Pressman (2005). Ingeniería de Software (Sexta Edición.). S.L: Mc Graw Hill.
24. Pressman & Ince (2005). Software engineering: a practitioner's approach: McGraw-Hill New York.
25. Pressman, R., & Ince, D. (2005). Software engineering: a practitioner's approach: McGraw-Hill New York.
26. Project Management Institute (2004). Guía de los fundamentos de la Dirección de Proyectos. (Guía del PMBOK). (Tercera ed.): Americano National Standart Institute. ANSI/PMI.
27. Project Management Institute (2000). Guía de los fundamentos de la Dirección de Proyectos. (Guía del PMBOK) (Segunda Edición.). Pennsylvania: Project Management Institute, Inc.
28. Rinia, E. (2007). Measurement and evaluation of interdisciplinary research and knowledge transfer., Leiden University, Belgium
29. Rumbaugh, J., Jacobson, I., & Booch, G. (2004). Unified Modeling Language Reference Manual, The: Pearson Higher Education.

## BIBLIOGRAFÍA REFERENCIADA

---

30. Senn, James A. (1990). Análisis y Diseño de Sistema de Información (Segunda Edición).
31. Stuwart & Gess, (2009). Herramientas y Técnicas de Gestión .
32. Yanexis; Correoso Martínez, Daymirelis (2007). Desarrollo Colaborativo de Software con la herramienta GForge. Facultad 10. Ciudad de la Habana Universidad de las Ciencias Informáticas. 75. p. [Disponible en: [http://bibliodoc.uci.cu/TD/TD\\_0890\\_07.pdf](http://bibliodoc.uci.cu/TD/TD_0890_07.pdf)]
33. Zapata & Julielsy, (2008). Uso y aplicación de las herramientas colaborativas en la educación y en la vida profesional con énfasis en el Weblog. Disponible.

## Índice de figuras y tablas

### Índice de figuras

FIGURA 1 ACTIVIDADES DEL SISTEMA DE INFORMACIÓN.....	19
FIGURA 2 ETAPAS DEL CICLO DE VIDA DE LOS SISTEMAS.....	22
FIGURA 3 OTORGAR PERMISO DE VER REPORTES. ....	53
FIGURA 4 VISUALIZAR REPORTES DE UN PROYECTO.....	54
FIGURA 5 PLUGIN-IN DE RIESGO.....	55

### Índice de tablas

TABLA 1 DIAGNÓSTICO REALIZADO POR LOS ESPECIALISTAS DE LA DIRECCIÓN TÉCNICA. ....	38
TABLA 2 DESCRIPCIÓN DEL GFORGE.....	39
TABLA 3 DESCRIPCIÓN DEL REDMINE . ....	40
TABLA 4 DESCRIPCIÓN DE DOTPROJECT.....	42
TABLA 5 DESCRIPCIÓN DEL PROJECT.NET.....	44
TABLA 6 DIAGNÓSTICO REALIZADO POR LOS ESPECIALISTAS DE LA DIRECCIÓN TÉCNICA. ....	45

**Anexo. Comparación entre Project.net, Redmine y DotProject.**

Característica		Nivel de cumplimiento de las Herramientas		
		ProjectNet (9.0.0)	Redmine (0.8)	DotProject (2.1.2 rc1)
Perfil del proyecto		Insuficiente	No	No
Gestionar calendario y cronograma		Suficiente	No	Suficiente
	Desarrollo del cronograma	Suficiente	Suficiente	Suficiente
	Gestionar tiempos muertos (vacaciones)	Suficiente	No	Suficiente
	Reporte de estado	Insuficiente	Insuficiente	Suficiente
	General del proyecto	Suficiente	Insuficiente	Suficiente
	Personal	Insuficiente	Suficiente	Suficiente
	Por componentes	No	Suficiente	No
	Por roles	No	Suficiente	No
	Por etapas	No	Suficiente	Suficiente
	Controlar los cambios realizados al cronograma	Insuficiente	Insuficiente	No Sé
	Diagramas de red, de barras, o de hitos	Suficiente	No	No
Modelar estructura de la organización		No	Insuficiente	No
	Facultad	Insuficiente	Insuficiente	No
	Polo	Insuficiente	Insuficiente	No
	Área temática	Insuficiente	Insuficiente	No
	Proyecto	Suficiente	Suficiente	No
	Subproyecto	Suficiente	Suficiente	No
	Obtener reportes a cada nivel y generales	Insuficiente	Suficiente	No
Gestionar tamaño y complejidad del producto		Insuficiente	No	No Sé
Gestionar riesgos		Insuficiente	Insuficiente	Insuficiente
	Identificar riesgo	Suficiente	Suficiente	Suficiente
	Impacto	Suficiente	Suficiente	Suficiente
	Tipo de riesgo (amenaza u oportunidad)	No	Suficiente	Insuficiente
	Probabilidad de ocurrencia	No	Suficiente	Suficiente
	Plan de respuesta (estado)	No	No	Suficiente
	Plan de mitigación (amenaza)	No	No	No
	Plan de aprovechamiento o	No	No	No

	mejora (oportunidad)			
	Responsable	No	Suficiente	Suficiente
	Matriz DAFO	No	No	No
	Diagrama Causa-Efecto	No	No	No
	<b>Clasificación</b>	No	No	No Sé
	Análisis cualitativo	No	No	Suficiente
	Análisis cuantitativo	No	No	Suficiente
	Planificación de las respuesta	No	No	Insuficiente
	Agrupar por categoria	No	Suficiente	No
	Matriz de probabilidad	No	No	No
<b>Gestionar costo</b>		<b>Insuficiente</b>	<b>No Sé</b>	<b>Insuficiente</b>
	Estimación	Insuficiente	No	Suficiente
	Preparación de presupuesto	Insuficiente	No	Insuficiente
	Control	Suficiente	No	No
	Valor Ganado	No	No	No
	Análisis de varianza	No	No	No
	Análisis de tendencia	No	No	No
<b>Gestionar alcance</b>		<b>Insuficiente</b>	<b>No</b>	<b>No</b>
	Gestionar objetivos	No	Suficiente	No
	Objetivos terminados	No	Suficiente	No
	Trazabilidad o dependencias	No	Insuficiente	No
	Planificación del alcance	No	No Sé	No
	Definición del alcance	No	No Sé	No
	Modelar ciclo completo de desarrollo	Suficiente	Insuficiente	No
	Fases	Suficiente	Insuficiente	No
	Hitos	Suficiente	Suficiente	No
	Iteraciones	Suficiente	No	No
	EDT o WBS	No	No	No
	Identificar productos entregables y el trabajo relacionado	Suficiente	No	No
	Estructurar y Organizar EDT	No	No	No
	Descomponer niveles superiores de EDT en componentes detallados de nivel inferior	No	No	No
	Desarrollar y asignar códigos de identificación a los componentes de la EDT	No	No	No
	Verificar que el grado de descomposición del trabajo es necesario y suficiente	No	No	No
	Verificación del alcance (garantizar compromisos con el alcance)	No	No	No

	Control de alcance (gestión de problemas/desviaciones y gestión de acciones correctivas)	No	No	No
<b>Gestionar revisiones</b>		<b>Insuficiente</b>	<b>No Sé</b>	<b>No</b>
	<b>Registrar (descripción, fecha de realización)</b>	No	No Sé	No
	Revisores	No	No Sé	No
	Secciones registradas	No	No Sé	No
	Responsable de modificar el plan	No	No Sé	No
	Tiempo invertido	No	No Sé	No
	Seguimiento	No	No Sé	No
	Resultado	No	No Sé	No
	Acciones correctivas	No	No Sé	No
	Cambios realizados	No	No Sé	No
	Comentarios	No	No Sé	No
<b>Gestionar tareas</b>		<b>No</b>	<b>Suficiente</b>	<b>Suficiente</b>
	Registrar tarea	Suficiente	Suficiente	Suficiente
	Asignar responsable	Suficiente	Suficiente	Suficiente
	Fecha de inicio y fin	Suficiente	Suficiente	Suficiente
	Esfuerzo	Suficiente	No	Suficiente
	Avance	Suficiente	Suficiente	Suficiente
	Campos personalizables	Suficiente	Suficiente	Suficiente
	Asignar Recurso	Suficiente	No	Suficiente
	Establecer dependencias (obligatorias, direccionales, externas) y restricciones entre ellas	Suficiente	Suficiente	Suficiente
	Método de diagramación por dependencia (PDM) o actividad en el nodo (AON)	No	No Sé	No
	Método de diagramación por flechas (ADM) o actividad en la flecha (AOA)	No	No Sé	No
<b>Gestionar acciones (tareas con nuevos campos)</b>		<b>Insuficiente</b>	<b>Suficiente</b>	<b>Suficiente</b>
	Tipo	Insuficiente	Suficiente	Suficiente
	Correctiva: para solucionar problemas o desviaciones ocurridas	No	Suficiente	Suficiente
	Preventiva: para solucionar problemas o desviaciones que pueden ocurrir	No	Suficiente	Suficiente
	Propósito	No	Suficiente	Suficiente
	Fecha de asignación	Suficiente	Suficiente	Suficiente

## ANEXOS

	Fecha de cumplimiento	Suficiente	Suficiente	Suficiente
	Responsable	Suficiente	Suficiente	Suficiente
Gestionar compromisos		Suficiente	No	No
	Actividades asociadas al compromiso	Suficiente	No	No
	Aceptación de actividades	Suficiente	No	No
	Control	Suficiente	No	No
	De cumplimiento	Suficiente	No	No
	De aceptación	Suficiente	No	No
Gestionar no conformidades		Suficiente	Suficiente	No
	Identificar	Insuficiente	Suficiente	No
	Asignar responsable	Insuficiente	Suficiente	No
	Control y seguimiento	Insuficiente	Suficiente	No
	Versión	Insuficiente	Suficiente	No
	Continuidad y discontinuidad	Insuficiente	No Sé	No
	Resolución de NC	Insuficiente	Suficiente	No
Gestionar problemas y desviaciones		Suficiente	No	No
	Identificar	Insuficiente	No	No
	Tipo	Insuficiente	No	No
	Descripción	Insuficiente	No	No
	Unidad (duración/costo/ complejidad)	Insuficiente	No	No
	Valor planeado	Insuficiente	No	No
	Valor real	Insuficiente	No	No
	Requerimientos afectados	Insuficiente	No	No
	Impacto	Insuficiente	No	No
	Criterio para decidir si es o no significativa	Insuficiente	No	No
	Origen	Insuficiente	No	No
	Causa	Insuficiente	No	No
	Asignar responsable	Insuficiente	No	No
	Control y seguimiento	Insuficiente	No	No
Gestionar recursos (humanos, materiales, tecnológicos)		Suficiente	No	Insuficiente
	Disponibilidad, capacidades	Suficiente	No	Suficiente
	Descripción	Suficiente	No	Suficiente
	Planificación de recursos	Suficiente	No	No
	Asignar recurso	Suficiente	No	Suficiente
	Control de recurso	Suficiente	No	Suficiente
	Curva de distribución de Recursos	No	No	No
	Adquisición y baja del recurso	No	No	Insuficiente
	RBS (estructura de desglose de recurso)	No	No	No

## ANEXOS

	Matriz RAM (asignación de responsabilidades)	No	No	No
	Gestionar interesados (stakeholder)	Insuficiente	No	Suficiente
	Identificar	Insuficiente	No	Suficiente
	Nombre y Apellidos	No	No	Suficiente
	Tipo (interno o externo)	No	No	Suficiente
	Necesidades, deseos, expectativas	No	No	Suficiente
	Obligaciones (compromisos)	No	No	Suficiente
	Enfoque del marco lógico	No	No	No
	árbol de problemas	No	No	No
	Matriz del marco lógico	No	No	No
	Matriz de trazabilidad	No	No	No Sé
	Procesamiento de diargama de GANTT, PERT, GERT	Suficiente	No	Suficiente
	Determinar ruta crítica	Suficiente	No	No
	Gestionar portafolio o carpeta de proyectos	Suficiente	No	No
	Alineado a la estrategia de la organización	Suficiente	No	No
	Tratar los proyecto como algo global	No	No	No
	Matrices de portafolios	No	No	No
	Diagrama Reckitt-Colman	No	No	No
	Tablas estratégicas	No	No	No
	Agrupar los proyectos en múltiples grupos o carteras	Suficiente	No	No
	Búsqueda de proyectos que cumplan determinadas características	No	No	No
	Localizar recursos donde importan	Suficiente	No	No
	Vincular las decisiones relacionadas con el portafolio a las directrices y objetivos estratégicos de la organización	No	No	No
	Para la correcta formación del portafolio se deben tener en cuenta los proyectos similares en cuanto a:	No	No	No
	Tamaño y complejidad	No	No	No
	Duraciones comparables	No	No	No
	Poca necesidad de recursos	No	No	No
	Desarrollados en ámbitos con tecnologías y disciplinas similares	No	No	No
	Versionado de los proyectos	Insuficiente	No	Insuficiente



## ANEXOS

	Agrupar funcionalidades, errores, peticiones, riesgos, recursos	Insuficiente	No	Insuficiente
Elementos de integrabilidad con otras herramientas		Insuficiente	No	Insuficiente
	Gestión de requisitos	No	No	No
	Gestión de errores	Insuficiente	No	Suficiente
	Gestión de configuración	No	Suficiente	No
	Gestión documental	Suficiente	Insuficiente	Suficiente
	Estimación	Insuficiente	No	No
	Servicios web u otras formas de integración	No Sé	Insuficiente	No
Gestión de conocimiento		Insuficiente	No	No
	Establecer temas	No	No	No
	Niveles de aceptación de los temas	No	No	No
	Matriz de Conocimiento	No	No	No
	Selección de personal x competencia	No	No	No
	Tabla de competencias	No	No	No
	Evaluación de competencias	No	No	No
Gestionar recursos humanos		Insuficiente	No	Suficiente
	Disponibilidad (si esta o no y condiciones), capacidades	Insuficiente	No	Suficiente
	Planificación de recursos	Suficiente	No	Suficiente
	Asignar recurso	Suficiente	Suficiente	Suficiente
	Control de recurso	Suficiente	No	Suficiente
	Adquisición y baja del recurso	Suficiente	No	Suficiente
	RBS (estructura de desglose de recurso)	No	No	No
	Matriz RAM (asignación de responsabilidades)	No	No	No
Gestionar elementos reutilizables		Insuficiente	No	No
	Identificar	No	No	No
	Nombre	No	No	No
	Descripción	No	No	No
	Propósito	No	No	No
	Características	No	No	No
	Fuente	No	No	No
	Asignar (Espacio de utilización)	No	No	No
	Identificar elementos reutilizables del proyecto	No	No	No
	Nombre	No	No	No
	Descripción	No	No	No
	Propósito	No	No	No

## ANEXOS

		Características	No	No	No
		Fuente	No	No	No
Gestionar adquisiciones			Insuficiente	No	No
	Identificar		No	No	No
		Nombre	No	No	No
		Descripción	No	No	No
		Propósito	No	No	No
		Características	No	No	No
		Fuente	No	No	No
	Fecha de la adquisición		No	No	No
	Asignar (Espacio de utilización)		No	No	No
Gestionar proveedores			Insuficiente	No	Suficiente
	Identificar		No	No	Suficiente
		Nombre	No	No	Suficiente
		Dirección	No	No	Suficiente
		Telefono, Email, Fax	No	No	Suficiente
	Identificar que se brinda por proveedor		No	No	Suficiente
		Detalles	No	No	Suficiente
		Facilidades	No	No	Suficiente
	Matriz de mercado		No	No	No

### Glosario de Términos

**Sistemas de Gestión:** estructura probada para la gestión y mejora continua de las políticas, los procedimientos y procesos de la organización.

**Gestión de Proyectos:** aplicación de conocimientos, habilidades, herramientas y técnicas a las actividades de un proyecto para satisfacer los requisitos del mismo.

**Tracker:** tipo de software con la función de un secuenciador para música, que permite añadir muestreos digitales o samples en listas de tiempos que se reparten en canales.

**Sistema:** conjunto de elementos dinámicamente relacionados formando una actividad para alcanzar un objetivo, operando sobre datos, energía o materia para proveer información.

**Proyecto:** planificación que consiste en un conjunto de actividades que se encuentran interrelacionadas y coordinadas, surge como respuesta a una necesidad, finaliza cuando se obtiene el resultado deseado, desaparece la necesidad inicial, o se agotan los recursos disponibles.

**Subsistema:** componente genérico que puede ser descompuesto en componentes específicos. Los componentes de un sistema pueden estar relacionado de diferentes formas: cada interrelación particular de componentes es una configuración.

**Plug-in:** complemento de una aplicación que se relaciona con otra para aportarle una función nueva y generalmente muy específica.

**Prototipo:** sistema que funciona es decir la primera versión, o iteración, de un sistema de información, es el modelo original.