

Universidad de la Ciencias Informáticas

Facultad 2



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

*Título: Sistema de Seguimiento y Control del Plan de  
Trabajo individual de los trabajadores del  
Centro de Telemática.*

**Autores:** Maité Benítez Ortiz.

Alexei Sánchez Figueroa.

**Tutores:** Ing. Bárbara Triana Morales.

Ing. Denys Buedo Hidalgo.

## DECLARACIÓN DE AUTORÍA

Declaramos que somos los únicos autores de este trabajo, por lo cual autorizamos a la Universidad de las Ciencias Informáticas (UCI) y a la Facultad (2) para que hagan el uso que estimen pertinentes con este trabajo.

Para que así conste firmo la presente declaración de auditoría a los \_\_\_\_ días del mes de junio del 2011.

X

Maité Benítez Ortiz

X

Alexei Sánchez Figueroa

X

Ing. Bárbara Triana Morales

X

Ing. Denys Buedo Hidalgo



*“...Todo hombre debe decidir una vez en la vida si se lanza a triunfar arriesgándolo todo, o se sienta a contemplar el paso de los triunfadores...”*

*Benedetti*

### AGRADECIMIENTOS

*A mis padres y a mi abuelita linda, por el cariño y el amor que me han brindado toda la vida, por ser mis principales educadores, el motivo de inspiración de llevar todos estos sueños adelante y poder siempre contar con ellos. Por hacer hasta lo imposible porque durante estos 5 años lejos de casa no me faltara nada, ni su cariño aún estando lejos, y por confiar en mí hasta en los momentos más difíciles.*

*A mi novio por constituir para mí un ejemplo a seguir y darme el aliento necesario para no retroceder ante las dificultades, por su amor y su apoyo, por ser mi mejor amigo, por malcriarme y quererme como soy.*

*A mis hermanas por su amor, por ser las cositas más chiquitas que tengo en la vida, por darme fuerzas para no defraudarlas nunca.*

*A mis tíos, Toni, Tita, Yeyi y Yamile, por apoyarme con su cariño incondicional y sus consejos, y por preocuparse por mí en cada momento durante toda mi carrera.*

*A los esposos de mis padres Jorge y Maricelis, por todo su apoyo y convertirse en segundos padres para mí.*

*A mi suegra por todo su apoyo en los momentos difícil, por su comprensión y su ayuda, por darme ánimo cuando más lo necesitaba, por haberse convertido en mi madre y mi amiga.*

*A mis tutores en especial a **Barbará** por ser nuestro faro y nuestra guía y a **mi compañero Alexeis** por su apoyo incondicional, y por aportar su granito de arena para que mi sueño se hiciera realidad. A mis amigos Marvis, Margarita, Addiarys, Rissel, Rosana y Karel por ser las mejores amigos del mundo, por su preocupación, su paciencia ante mis cambios de ánimo y por soportar todas mis malcriadeces, por eso y mucho más...gracias.*

*Maité*

*A mi familia en especial a mi mamá y mi papá por el apoyo que siempre me brindaron en todo momento.*

*A mi novia por estar siempre ahí cuando la necesitaba, a mi hermanita que espero que siga este ejemplo que le pretendo dar.*

*A mis abuelos por todo el amor que siempre me han brindado y por hacerme un hombre de bien.*

*A mis tutores y compañera Maité por ayudarme a conseguir este sueño mío y de mi familia.*

*A todos mis amigos que siempre me apoyaron y ayudaron cuando más lo necesitaba.*

*A todos los profesores de la facultad que de una forma u otra aportaron un granito de arena para poder llevar a cabo el desarrollo de esta tesis.*

*Muchas gracias a todos.*

*Alexei*

*Dedico este trabajo a mis padres y a mi abuela por alumbrarme el camino, por ser mis guías en todo momento, por confiar en mí y lograr a pesar de los obstáculos que haya terminado mis estudios y pudiera superándome para ser alguien en la vida.*

*Maité*

*Le dedico este trabajo y mi esfuerzo a toda mi familia, mi mamá, mi papá, mi hermana, mi novia porque siempre estuvieron presentes ante cualquier problema. Gracias por enseñarme el camino correcto para lograr este gran sueño.*

*Alexei*

### **RESUMEN**

En la Universidad de las Ciencias Informáticas (UCI) se genera un gran cúmulo de información como parte del proceso de control y seguimiento de los Planes de Trabajo de cada uno de sus trabajadores, por lo que los procesos de recuperación, consulta, filtrado y gestión de la información se hacen engorrosos.

La UCI está compuesta por centros productivos en consecuencia con las características de ser una institución desarrolladora de soluciones informáticas. Uno de los centros de desarrollo de software de la universidad es el Centro de Telemática.

El Centro de Telemática cuenta con pocos años de fundado, como resultado de su poca experiencia algunos procesos no están identificados o se implementan mal. Actualmente para la confección de los Planes de Trabajo se utilizan vías lentas y poco factibles, lo que dificulta la obtención de resultados relevantes en tiempo y forma.

Con el desarrollo de la presente investigación se logró implementar el Sistema de Seguimiento y Control de los Planes de Trabajo del Centro de Telemática, posibilitando de esta forma que se lleve un mejor control de las tareas a realizar por los trabajadores, permitiendo que los directivos a cargo puedan no solamente asignar tareas sino también verificar el cumplimiento de las mismas, obtener reporte de los Planes de Trabajo individuales, enviar notificaciones sobre la realización de una tarea. Para el desarrollo de la solución se utilizó la metodología de desarrollo de software XP, el lenguaje de programación Java y Eclipse como Entorno de Desarrollo Integrado.

### **Palabras claves:**

Control, plan de trabajo, seguimiento, tareas.

---

<b>ÍNDICE</b>	
<b>INTRODUCCIÓN</b> .....	4
<b>CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA</b> .....	9
<b>Introducción</b> .....	9
<b>1.1. Fundamentación del Tema</b> .....	9
1.1.1. Control de recursos humanos.....	9
1.1.2. Plan de Trabajo.....	10
<b>1.2. Sistema de gestión</b> .....	11
<b>1.3. Tendencias en el uso de sistemas de gestión de Planes de Trabajo</b> .....	11
<b>1.4. Análisis de soluciones existentes</b> .....	12
1.4.1. BASECAMP .....	12
1.4.2. Planificador: Sistema para la confección de planes de trabajo (Centro de Estudios de Ciencias de la Educación Superior Universidad de Pinar del Río).....	12
1.4.3. GESPRO.....	13
<b>1.5. Técnicas de Inteligencia Artificial</b> .....	14
<b>1.6. Metodologías, lenguajes y herramientas de Desarrollo</b> .....	15
1.6.1. Metodología de desarrollo de software .....	16
1.6.2. Lenguaje de Modelado .....	17
1.6.3. Procedimiento de Modelado de Procesos IDEF0.....	18
1.6.4. Lenguaje de Programación.....	19
1.6.5. Herramienta CASE .....	19
1.6.6. Framework de Desarrollo .....	20
1.6.7. Entorno de Desarrollo Integrado .....	22
1.6.8. Sistema Gestor de Base de Datos .....	23
1.6.9. Servidor Web .....	23
1.6.10. Sistema de Control de Versiones .....	24
1.6.11. Conclusiones parciales.....	25
<b>CAPÍTULO 2: CARACTERÍSTICAS DEL SISTEMA. EXPLORACIÓN Y PLANIFICACIÓN</b> .....	26



<b>Introducción</b> .....	26
<b>2.1. Situación Problemática</b> .....	26
<b>2.2. Modelo de proceso del negocio</b> .....	27
2.2.1. Identificar actividades básicas para el trabajador .....	28
2.2.2. Designar fechas de cumplimiento: .....	30
2.2.3. Revisar actividades:.....	30
<b>2.3. Propuesta de solución</b> .....	30
<b>2.4. Especificación de los requisitos del software</b> .....	31
2.4.1. Requisitos no funcionales .....	31
<b>2.5. Personas relacionadas con el Sistema</b> .....	33
<b>2.6. Fase de Exploración</b> .....	33
2.6.1. Historias de Usuario .....	33
<b>2.7. Fase de planificación.</b> .....	40
2.7.1. Estimación de esfuerzo por HU. ....	40
2.7.2. Plan de iteraciones. ....	41
2.7.3. Plan de duración de iteraciones. ....	42
2.7.4. Plan de entregas .....	43
<b>2.8. Conclusiones Parciales</b> .....	43
<b>CAPÍTULO 3: DISEÑO DEL SISTEMA</b> .....	44
<b>Introducción</b> .....	44
<b>3.1. Diseño del Sistema</b> .....	44
3.1.1. Patrones Aplicados .....	44
3.1.2. Estilo arquitectónico.....	46
3.1.3. Tarjetas CRC (Clase-Responsabilidad-Colaboración) .....	47
3.1.4. Prototipos de interfaz de usuario .....	50
3.1.5. Diseño de la Base de Datos .....	55
<b>CAPÍTULO 4: IMPLEMENTACIÓN Y PRUEBA</b> .....	58
<b>Introducción</b> .....	58
<b>4.1. Implementación</b> .....	58

4.1.1.	Algoritmo de Inteligencia Artificial. Primero el mejor .....	58
4.1.2.	Tareas de ingeniería .....	59
4.1.3.	Diagrama de Despliegue .....	63
<b>4.2.</b>	<b>Validación de la Propuesta</b> .....	<b>64</b>
4.2.1.	Pruebas Unitarias .....	64
4.2.2.	Pruebas de Aceptación.....	65
4.2.3.	Resultados de las pruebas de aceptación para cada iteración de prueba .....	66
<b>CONCLUSIONES</b>	.....	<b>67</b>
<b>REFERENCIAS BIBLIOGRÁFICAS</b>	.....	<b>68</b>
<b>BIBLIOGRAFÍA</b>	.....	<b>71</b>

### INTRODUCCIÓN

La información en la sociedad ha estado creciendo de manera exponencial, las diferentes empresas y organizaciones generan grandes cantidades de datos que se procesan con el objetivo de lograr una mayor productividad y eficiencia en su trabajo. Este incremento trae consigo la aparición de novedosos y sofisticados sistemas informáticos que gestionan en las empresas, tanto su información, como el desempeño de sus trabajadores.

Cuba no está ausente a este avance tecnológico debido al esfuerzo que ha venido realizando el gobierno al llevar a cabo la informatización de los diferentes sectores de la economía y la sociedad. Este proceso genera cambios significativos en el funcionamiento interno de las entidades y promueve la necesidad de llevar el control y seguimiento estricto del plan de trabajo de cada uno de los trabajadores, posibilitando una mejor organización y perfeccionamiento del capital humano de la empresa.

El control y seguimiento del plan de trabajo individual permite a una organización obtener información sobre actividades pasadas y en curso que se pueden usar como base para ajustar la programación de las mismas y reorientarlas para la planificación futura. El análisis de dicho documento permite realizar comparaciones de lo realizado durante un período determinado por un trabajador y asignar la evaluación de éste en el intervalo de tiempo evaluado.

En el país las máximas autoridades de los órganos, organismos, organizaciones y demás entidades deben integrar a su Sistema de Control Interno, el proceso de planificación, los objetivos y planes de trabajo, para relacionarlo con los procesos, actividades y operaciones en el interés de asegurar el cumplimiento de su misión y de las disposiciones legales que le competen.[1]

Las Universidades del país cuentan con una estructura compuesta por el personal del rectorado, las facultades, los centros, las sedes universitarias, las becas estudiantiles entre otras. El equipo rectoral de cada universidad es el encargado de confeccionar los Planes de Trabajo de los trabajadores, resumiendo las propuestas de los dirigentes pertinentes en cada una de sus áreas.[2]

La Universidad de las Ciencias Informáticas (UCI) como empresa desarrolladora de software también necesita llevar un control sobre todos los trabajadores para así mejorar y explotar al máximo la productividad de los mismos. Sin una planificación, un seguimiento y una evaluación eficaz, sería imposible juzgar si el trabajo va en la dirección correcta.

En la UCI se genera un gran cúmulo de información como parte del proceso de control y seguimiento de los planes de trabajo de cada uno de sus trabajadores, por lo que los procesos de recuperación, consulta, filtrado y procesamiento se hacen engorrosos.

La UCI está compuesta por centros productivos en consecuencia con las características de ser una institución desarrolladora de soluciones informáticas. La facultad 2 se encuentra dividida en dos centros de desarrollo de software: Centro Informatización de la Seguridad Ciudadana y Centro de Telemática.

El Centro de Telemática cuenta con pocos años de fundado. Existen algunos procesos que no están identificados o se implementan mal. Uno de los principales problemas es que no se realiza un control estricto de las actividades realizadas por los trabajadores del centro.

Las actividades a realizar por cada trabajador deben estar incluidas en el plan de trabajo así como el tiempo de duración de cada una de ellas y el área temática a la que pertenezca ya sea productiva, investigativa, docente o de extensión universitaria. Los planes de trabajo son realizados por los jefes de cada departamento del centro y se aprueban en conjunto con el trabajador a evaluar.

Actualmente para la confección de los planes de trabajo se utilizan vías lentas y poco factibles, lo que dificulta la obtención de resultados relevantes en tiempo y forma. Se realiza una inadecuada gestión de los mismos pues no son utilizados para realizar la evaluación de desempeño, que no es más que el proceso de asignar la evaluación trimestral del profesor en las categorías adecuado o superior.

El plan de trabajo no se utiliza para realizar un control periódico sobre las actividades a realizar por cada trabajador. Según las especificidades de cada profesor se deben tener en cuenta condiciones que son de cumplimiento obligatorio, por ejemplo: Un profesor instructor debe recibir 4 cursos de posgrado anualmente. Estas condiciones aunque están definidas por el centro y por la Universidad, no están a la vista de los directivos y en algunas ocasiones no se tienen en cuenta en el momento de la planificación de las tareas.

A la hora de asignar una actividad los directivos del centro pierden demasiado tiempo pues tienen que seleccionar entre todos los trabajadores del centro aquellos que estén más capacitados para ejercer una tarea determinada, de ahí que resulte problemático la selección del personal que pudiera cumplir satisfactoriamente una labor determinada.

Debido a lo expuesto anteriormente se plantea el siguiente **problema a resolver**: ¿Cómo convertir el plan de trabajo de los profesores del Centro de Telemática en un mecanismo efectivo para el control y seguimiento de las tareas planificadas?

Teniendo como **objeto de estudio** de la investigación el proceso de confección, control y seguimiento del plan de trabajo individual de los trabajadores, el **campo de acción** se enmarca en los sistemas que gestionan información relacionada con los Planes de Trabajo individuales de los trabajadores.

Por tanto el **objetivo general** de esta investigación es: desarrollar un sistema informático que permita gestionar el plan de trabajo individual de los trabajadores del Centro de Telemática.

De este objetivo general se derivan los siguientes **objetivos específicos**:

1. Desarrollar el marco teórico conceptual de la investigación.
2. Desarrollar el Modelo de Diseño de un sistema informático que permita elaborar y controlar el Plan de trabajo individual de los trabajadores del Centro de Telemática.
3. Desarrollar el Modelo de Implementación de un sistema informático que permita elaborar y controlar el Plan de trabajo individual de los trabajadores del Centro de Telemática.
4. Aplicar técnicas de inteligencia artificial para la asignación de tareas a los trabajadores.
5. Validar el sistema propuesto.

En vista de lograr el cumplimiento de los objetivos, han surgido las siguientes **preguntas científicas**:

- ¿Existen sistemas de gestión para los planes de trabajo? ¿Qué aportes pueden ofrecer a la propuesta de solución planteada?
- ¿Cuál es la metodología de desarrollo de software idónea para guiar el desarrollo de la aplicación final?
- ¿Cuáles herramientas serán necesarias para modelar y construir el sistema?
- ¿Qué lenguaje de programación sería el más eficiente para dar solución al problema?
- ¿Cuál framework sería el más eficaz para guiar a los desarrolladores y dar una solución al problema planteado?
- ¿Qué técnicas de inteligencia artificial podrían ser utilizadas para la gestión de actividades?
- ¿Qué actividades se realizan en el Centro de Telemática para la gestión de los planes de trabajo de los trabajadores y quiénes son los responsables?

Las siguientes **tareas de investigación** están encaminadas a dar solución a las interrogantes antes planteadas.

- Estudio de conceptos y leyes relacionadas con la gestión de los Recursos Humanos (RRHH).
- Estudio de los procesos que se llevan a cabo en las Universidades del mundo, del país y en la UCI para la confección, control y seguimiento de los Planes de Trabajo.
- Realización de un estudio de los sistemas de gestión de RRHH existentes en el mundo y el país para obtener una visión general de la experiencia alcanzada, hasta el momento, en este campo.
- Estudio sobre las herramientas, metodología de desarrollo de software, lenguaje de programación, framework y plataforma a utilizar en el desarrollo del sistema.
- Identificar las técnicas de inteligencia artificial que serán aplicadas en el desarrollo del sistema para determinar el personal más calificado para realizar una determinada actividad.
- Realizar el Diseño del Sistema.
- Realizar la implementación del Sistema.
- Verificación del producto realizado para comprobar el correcto desarrollo de las funcionalidades requeridas por el cliente.

En todo el trayecto de la investigación se utilizarán **métodos científicos** para lograr caracterizar a fondo el objeto de estudio así como garantizar el conocimiento del estado del arte, su evolución y relación con otros fenómenos. Los métodos científicos a utilizar son:

- **Histórico-Lógico:** este método se utiliza con el fin de estudiar sistemas informáticos que confeccionen, controlen y gestionen Planes de Trabajos.
- **Analítico-Sintético:** a través de este método se analiza la bibliografía disponible para realizar una investigación lo más completa posible sobre el proceso de confección, gestión y control de los Planes de Trabajo.
- **Entrevista:** se utiliza en la investigación para precisar el problema a resolver, las necesidades del cliente y los procesos que se llevan a cabo actualmente para la gestión de los Planes de Trabajo del Centro de Telemática.

El documento está estructurado en cuatro capítulos:

**Capítulo1:** Fundamentación Teórica. Este capítulo incluye el estado del arte del tema tratado, a nivel nacional e internacional. Se realiza un estudio de las tecnologías y metodologías que pueden ser utilizadas en la solución informática y se seleccionan las más idóneas.

**Capítulo2:** Características del Sistema. Exploración y Planificación. Se describe todo el proceso de negocio a través de la notación IDEF0, así como se define lo que debe hacer el sistema a partir de las funcionalidades requeridas y las restricciones impuestas por el cliente.

**Capítulo3:** Diseño del sistema: Se describen las clases del diseño utilizando las tarjetas CRC (Clase-Responsabilidad-Colaboración). Se define la arquitectura base y los patrones de diseño a utilizar. Por último se tiene el diseño de la Base de Datos del sistema.

**Capítulo4:** Implementación y Prueba. Se describe todo el proceso de implementación del sistema. Se realizan los casos de prueba al sistema implementado y se muestran los resultados de cada una de las iteraciones de pruebas realizadas.

## **CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA**

### **Introducción**

En el presente capítulo se realiza una investigación relacionada con los diferentes sistemas de gestión de los planes de trabajo en el mundo, el país y la Universidad. Se exponen una serie de definiciones y explicaciones que fundamentarán la utilización de las herramientas de desarrollo que permitirán dar solución al problema planteado.

### **1.1. Fundamentación del Tema**

Las organizaciones están compuestas por personas o grupos, en vista a conseguir ciertos fines y objetivos, por medio de tareas diferenciadas que se procura que estén racionalmente coordinadas y dirigidas. En la actualidad las organizaciones sufren cambios, afectando la forma en la que se gestionan los recursos humanos, por lo que el éxito de toda organización depende de una serie de factores, en su mayoría, referidos a la actividad de Gestión de Recursos Humanos; conllevando a la necesidad de adquirir nuevas competencias en cuanto a cómo se controlan los RRHH en tiempos de incertidumbre e inestabilidad.

#### **1.1.1. Control de recursos humanos**

La palabra control tiene muchas connotaciones y su significado depende de su función o del área específica en que se aplique. Puede ser la función administrativa y gerencial de control. En este caso, el control hace parte del proceso administrativo, conjuntamente con la planeación, la organización y la dirección. El control no consiste solo en verificar si todo va de acuerdo con el plan adoptado, las instrucciones emitidas y los principios establecidos, tiene por objetivo señalar las fallas y los errores para rectificarlos y evitar reincidir en ellos. Se aplica a todo: cosas, personas y actos.[3]

Con el control y registro del personal, se trata de asegurar que las diversas unidades de la organización marchen de acuerdo con lo previsto. Cuanto mayor sea la organización y más descentralizada sea su estructura, mayor será la necesidad de controlar los RRHH.



## 1.1.2. Plan de Trabajo

Un **Plan de Trabajo** es una herramienta que permite ordenar y sistematizar información relevante para realizar un trabajo. Esta especie de guía propone una forma de interrelacionar los RRHH, financieros, materiales y tecnológicos disponibles. Como instrumento de planificación, el Plan de Trabajo establece un cronograma, designa a los responsables, marca metas y objetivos. El Plan de Trabajo suele ser válido para un determinado período de tiempo. De esta manera, las acciones que propone deben desarrollarse en un cierto plazo y los objetivos tienen que ser cumplidos antes de una fecha límite. Al concluir un plan de trabajo este es remplazado por uno nuevo.[4]

Los Planes de Trabajos deben ser:

- Adecuados en el tiempo: Se pueden cumplir en un período razonable de tiempo.
- Flexibles: Se pueden modificar cuando se presenten situaciones inesperadas.
- Factibles: Deben ser reales y posibles de lograr.
- Convenientes: Su cumplimiento debe apoyar los propósitos y misiones de la organización.

Es posible caracterizar el Plan de Trabajo en múltiples tipos:

- Según la clase del Plan: objetivos, políticas a seguir, procedimientos, métodos, programas y presupuestos.
- Según los fines del Plan: nuevos, vigentes y correctivos.
- Según el uso del plan: para utilizarse una sola vez o varias veces.[5]

En la Universidad de las Ciencias Informáticas la planeación se concibe como un proceso permanente, orientado al cumplimiento de los objetivos que se tracen en cada una de las áreas de resultados claves: docencia, investigación, producción, extensión universitaria, entre otras. Para cumplir con estos objetivos la Universidad requiere planificar cada una de las actividades a realizar por los trabajadores en cada una de las áreas.

El proceso de planeación se concreta con la elaboración del Plan de Trabajo, el cual hace explícitas las actividades a desarrollar por los trabajadores para dar cumplimiento a los objetivos trazados.

## **1.2. Sistema de gestión**

Un sistema de gestión es una estructura probada para la gestión y mejora continua de las políticas, los procedimientos y procesos de la organización.[6]

Permite cumplir los objetivos de la organización mediante una serie de estrategias trazadas por la misma, que incluyen la optimización de procesos, el enfoque centrado en la gestión y el pensamiento disciplinado. Las empresas del siglo XXI se enfrentan constantemente a retos y cambios, como son la rentabilidad, competitividad, velocidad de los cambios, capacidad de adaptación, el bienestar de todos los empleados de la empresa y el cumplimiento en tiempo de todas las actividades.

Equilibrar estos y otros requisitos empresariales puede constituir un proceso difícil. Es aquí donde entran en juego los sistemas de gestión, al permitir no solamente aprovechar y desarrollar el potencial existente en la organización sino también renovar constantemente sus estrategias y operaciones.

## **1.3. Tendencias en el uso de sistemas de gestión de Planes de Trabajo**

En la actualidad, la tendencia de gestionar lo humano y las prácticas que visibilizan esta tendencia tienen el propósito de incrementar la productividad y competitividad.

En el momento actual se vienen imponiendo diferentes perspectivas teóricas que hablan de la gestión humana no sólo como un proceso necesario, sino como el centro mismo de la organización.

La gestión humana debe posibilitar el establecimiento de una estrategia que confíe en las personas como fuente de ventaja competitiva sostenida, la implementación de una cultura gerencial que comparta y defienda esta creencia, que entienda las implicaciones del capital humano sobre los problemas del negocio y pueda modificar el sistema de recursos humanos para solucionar esos problemas.[7]

Toda organización desea culminar sus metas con éxito, para ello es necesaria la aplicación correcta de su planificación estratégica previamente desarrollada por la dirección general de la empresa.

En la actualidad utilizar sistemas que gestionen los Planes de Trabajo de todos los empleados de la empresa o institución ofrece la posibilidad de trabajar en base al ser humano que está inmerso en la organización, en sus potencialidades y capacidades, generando un valor agregado y un aspecto diferenciador de cada organización que la haga más productiva.

## 1.4. Análisis de soluciones existentes

Luego de hacer un estudio preliminar del negocio se decidió investigar soluciones informáticas en el mundo y en el país que se ajustaran a la solución del problema.

### 1.4.1. BASECAMP

Es una herramienta de gestión de proyectos muy completa que incluye gestión de actividades, seguimiento y reporte de tiempos, gestión documental, RSS<sup>1</sup>, foros de comunicación, diseño de templates, calendarios, notificaciones, etc. Diseñado para gestionar proyectos de forma colaborativa, que brinda al grupo de trabajo las herramientas necesarias sin importar lo lejos que se encuentren los miembros que lo componen.[8]

Por otra parte es una herramienta sencilla de usar, la información se encuentra centralizada, gracias a la sindicación de contenidos por RSS, a las alertas por email o a la posibilidad de sincronizar el calendario con iCal<sup>2</sup>, el seguimiento del proyecto es mucho más sencillo y siempre estás al día, por último se puede decir que es una potente herramienta de comunicación y coordinación interna. [9]

Como desventajas presenta:

La herramienta, cuenta solamente con algunas opciones básicas de personalización. Además, sólo está disponible en lenguaje inglés. En los mensajes hay que usar Textile, una opción mucho más pobre que el HTML (lenguaje de marcado de hipertexto). Para que los usuarios puedan subir y compartir ficheros se requiere de un hosting<sup>3</sup> adicional con FTP<sup>4</sup>. Diseñado para Mac<sup>5</sup>.

### 1.4.2. Planificador: Sistema para la confección de planes de trabajo (Centro de Estudios de Ciencias de la Educación Superior Universidad de Pinar del Río)

Sistema para la confección de planes de trabajo que posibilita que coexistan varios usuarios, del tipo administrador o los encargados de diferentes áreas, permite realizar una gestión de las actividades a publicarse en el Plan de Trabajo de un mes determinado, permitiéndoles a los usuarios organizarlas de

---

<sup>1</sup>**RSS** son las siglas de *Really Simple Syndication*, un formato XML para indicar o compartir contenido en la web.

<sup>2</sup>**iCal** es una aplicación de calendario personal hecha por Apple Inc.

<sup>3</sup>**Hosting** significa poner una página web en un servidor de Internet para que esta pueda ser vista en cualquier lugar del mundo entero con acceso al Internet.

<sup>4</sup>**FTP** protocolo de red para la transferencia de archivos entre sistemas conectados a una red.

<sup>5</sup>**Mac** Serie de computadores personales de Apple.

forma tal que no existan coincidencias. Los planes generados pueden ser exportados a ficheros PDF<sup>6</sup> o imprimirse normalmente.[10]

Entre las principales desventajas que presenta este sistema se encuentran:

- No permite conocer el avance de las actividades.
- Solamente se gestionan de los usuarios su nombre y el área en la que laboran.
- No existe forma de registrar descripciones sobre las actividades.
- No se describe detalladamente el rango de días para realizar la actividad, solamente se conoce la fecha tope para realizarla.
- No cuenta con un mecanismo que posibilite notificar a los trabajadores sobre la realización de una tarea.

### **1.4.3. GESPRO**

Es un Paquete para la Gestión de Proyectos desarrollado por la Universidad de las Ciencias Informáticas UCI y comercializable desde las empresas comercializadoras asociadas a la Universidad. [11]

Este paquete incluye:

- Paquete de herramientas de gestión de proyectos en la organización (módulo básico)
- Subsistemas de dirección integrada de proyectos.
- Subsistemas de gestión de reportes dinámicos.
- Subsistemas para la gestión documental y el control de versiones.
- Subsistema de autenticación y seguridad.
- Subsistema de noticias y trabajo colaborativo y la vigilancia tecnológica.
- Subsistema de comunicación con correo y la mensajería.
- Subsistema para análisis estadísticos y monitoreo.
- Paquete de servicios de soporte y acompañamiento.
- Paquete de servicios de salvadas automáticas de recuperación de la información.
- Paquete de cursos de formación avanzados en Gestión de Proyectos (Maestría en Gestión de Proyectos).

---

<sup>6</sup>PDF formato de almacenamiento de documentos, desarrollado por la empresa Adobe Systems.

De las soluciones encontradas dos de ellas son herramientas de gestión de proyecto y en el caso de Planificador es el que más se asemeja a la solución del problema, pero no satisface las necesidades existentes en el Centro de Telemática con los Planes de Trabajo. Planificador es un sistema solamente para la confección de los Planes de Trabajo que no permite llevar un seguimiento sobre la realización de las tareas.

### 1.5. Técnicas de Inteligencia Artificial

La **inteligencia artificial** es una de las áreas de la informática que se encarga de crear software y hardware con comportamientos inteligentes. Trata de conseguir que un ordenador simule en cierta manera la inteligencia de los seres humanos para ello utiliza una serie de técnicas para lograr simular el razonamiento humano. En la actualidad la inteligencia artificial se ha aplicado a un sin número de actividades realizadas por las personas, entre las que se destacan: aplicaciones en la robótica, técnicas de aprendizaje y gestión del conocimiento.

Una **técnica de inteligencia artificial** es un método que utiliza conocimiento representado.[12]Una técnica de inteligencia artificial ayuda a acotar las posibilidades de solución para un problema determinado. Son utilizadas en diversas situaciones para corregir variados errores.

Entre las principales técnicas de inteligencia artificial se encuentran:

- Sistemas Basados en Conocimiento.
- Sistemas Expertos.
- Minería de Datos.
- Redes Semánticas.
- Redes Neuronales Artificiales.
- Métodos Heurísticos

La búsqueda de la inteligencia artificial difiere de la búsqueda convencional sobre estructuras de datos, esencialmente en que se busca en un espacio problema. Se busca un camino que conecte la descripción inicial del problema con una descripción del estado deseado para el problema, es decir, el problema resuelto. Este camino representa los pasos de solución del problema.[13]

Existen diferentes métodos de búsqueda como son la búsqueda a lo ancho y la búsqueda en profundidad, también denominados métodos ciegos. Estos métodos en general son muy ineficientes debido al

crecimiento del tiempo y el espacio que requieren. Otros tipos de métodos serían los de búsqueda heurística o búsqueda informada estos disponen de alguna información que les indique la proximidad de cada estado a un estado objetivo.

### **Búsqueda heurística**

Los métodos de búsqueda heurísticas están orientados a reducir la cantidad de búsqueda requerida para encontrar una solución.[13] También son conocidos como métodos fuertes ya que incorporan conocimiento heurístico. Un conocimiento heurístico no es más que un conjunto de reglas que evalúan la posibilidad de que una búsqueda va en la dirección correcta.[14]

### **Algoritmo de búsqueda Primero el Mejor**

El método de búsqueda Primero el Mejor representa una forma de combinar las ventajas que representan tanto la búsqueda primero en anchura como el primero en profundidad en un solo método.[15] La búsqueda primero en profundidad tiene la ventaja de que permite encontrar una solución sin tener que expandirse completamente por todas las ramas, la búsqueda primero en anchura presenta la ventaja de que no queda atrapada en callejones sin salida. Una forma de combinar ambas ventajas puede consistir en seguir un único camino cada vez, y cambiarlo cuando alguna ruta parezca más prometedora que la que se está siguiendo en ese momento.[15]

El método de inteligencia artificial Primero el Mejor será utilizado para facilitar el trabajo de los directivos del centro y ahorrar tiempo a la hora de asignar una actividad, de tal manera que cuando se tenga una actividad y no se sepa a qué trabajador o grupo de trabajadores asignárselas el sistema será capaz de proponer las personas disponibles y más idóneas para efectuar la tarea.

### **1.6. Metodologías, lenguajes y herramientas de Desarrollo**

Para lograr que el sistema cumpla con los requisitos necesarios se hace importante realizar una buena selección de las metodologías, lenguajes y herramientas de desarrollo a utilizar en la elaboración de un sistema que cubra los procesos que se desarrollan, en el presente capítulo se hará referencia a las principales características y facilidades que brinda cada uno de los elementos seleccionados.

## 1.6.1. Metodología de desarrollo de software

Las metodologías de desarrollo de software son un conjunto de procedimientos, técnicas y ayudas a la documentación para el desarrollo de productos software.[16]Las metodologías de desarrollo se clasifican en dos grupos; las tradicionales o robustas y las ágiles, a continuación se explican en qué consiste cada una de ellas:

### Metodologías robustas o tradicionales

Son aquellas con mayor énfasis en la planificación y control del proyecto, en especificación precisa de requisitos y modelado. Estas metodologías tradicionales imponen una disciplina de trabajo sobre el proceso de desarrollo del software, con el fin de conseguir un software más eficiente. Para ello, se hace énfasis en la planificación total de todo el trabajo a realizar y una vez que está todo detallado, comienza el ciclo de desarrollo del producto software. Se centran especialmente en el control del proceso, mediante una rigurosa definición de roles, actividades, artefactos, herramientas y notaciones para el modelado y documentación detallada.[17]

### Metodologías ágiles

Su objetivo es esbozar los valores y principios que deberían permitir a los equipos desarrollar software rápidamente, respondiendo a los cambios que puedan surgir a lo largo del proyecto. Se pretende ofrecer una alternativa a los procesos de desarrollo de software tradicionales, caracterizados por ser rígidos y dirigidos por la documentación que se genera en cada una de las actividades desarrolladas.[18]

Principales características de las metodologías ágiles:

- Se encarga de valorar al individuo y las iteraciones del equipo más que a las herramientas o los procesos utilizados.
- Se hace mucho más importante crear un producto software que funcione que generar mucha documentación.
- El cliente está en todo momento colaborando en el proyecto.

### Extreme Programming (XP)

Es una metodología ágil centrada en potenciar las relaciones interpersonales como clave para el éxito en el desarrollo de software, promoviendo el trabajo en equipo, preocupándose por el aprendizaje de los desarrolladores, y propiciando un buen clima de trabajo. XP se basa en realimentación continua entre el cliente y el equipo de desarrollo, comunicación fluida entre todos los participantes, simplicidad en las soluciones implementadas y coraje para enfrentar los cambios. [19]

La metodología XP se basa en:

- Pruebas Unitarias: se basa en las pruebas realizadas a los principales procesos, de tal manera que se adelanta en algo hacia el futuro, se puede hacer pruebas de las fallas que pudieran ocurrir. Es como si se adelantara a obtener los posibles errores.
- Programación en pares: una particularidad de esta metodología es que propone la programación en pares, la cual consiste en que dos desarrolladores participen en un proyecto en una misma estación de trabajo. Cada miembro lleva a cabo la acción que el otro no está haciendo en ese momento.

En la presente investigación se decide utilizar XP como metodología de desarrollo de software ágil ya que se cuenta con un equipo de tan solo dos personas y el tiempo de desarrollo es corto. Además por ser sencilla, tanto en su aprendizaje como en su aplicación, reduciéndose así los costos de implantación en el equipo de desarrollo. Por otra parte, las características de los proyectos para los cuales las metodologías ágiles han sido especialmente pensadas se ajustan perfectamente a la problemática. El cliente guiará la toma de decisiones y estará siempre informado del avance del desarrollo. En otras metodologías contar con el cliente es una ventaja, para XP esto sería una precondition, la colaboración entre ambos resolvería una gran cantidad de dudas y serviría para decidir qué hacer en cada momento, siempre en función de los intereses del negocio.

### 1.6.2. Lenguaje de Modelado

**UML** es un lenguaje gráfico para especificar, construir y documentar los artefactos que modelan un sistema.[20]UML fue diseñado para ser un lenguaje de modelado de propósito general, por lo que puede utilizarse para especificar la mayoría de los sistemas basados en objetos o en componentes, y para



modelar aplicaciones de muy diversos dominios de aplicación (telecomunicaciones, comercio, sanidad, etc.) y plataformas de objetos distribuidos.[20]

Se decidió utilizar UML ya que proporciona ventajas en la representación del ciclo de vida de un software y de los artefactos específicos del Proceso Unificado de Desarrollo del Software además de permitir una comunicación sencilla y rápida entre desarrolladores y clientes del software.

No se necesitan conocimientos profundos de Ingeniería del Software para que los clientes comprendan el objetivo y función del software, de modo que rápidamente pueden expresar su conformidad con el producto o las nuevas mejoras que desean ver introducidas.

### **1.6.3. Procedimiento de Modelado de Procesos IDEF0**

IDEF (Integrated Definition for Function Modelling): IDEF es una familia de técnicas de modelado sencilla, que ofrecen una perspectiva integrada para representar y modelar procesos y estructuras de datos. Sus inicios se remontan a la necesidad de las Fuerzas Armadas Estadounidenses por mejorar sus operaciones de producción. La familia IDEF, consiste en un gran número de técnicas, entre las cuales se destaca IDEF0 e IDEF3, que son aquellas relacionadas con los procesos de negocio. [20]

La técnica IDEF0, está diseñada para modelar las decisiones, acciones y actividades de una organización u otro sistema, y representa la perspectiva funcional de modelado. Es considerada una técnica sencilla pero poderosa, ampliamente usada en la industria durante la etapa de análisis en la reingeniería de procesos. Permite identificar apropiadamente los procesos y sus interfaces así como elaborar los documentos que permitan su control en cualquiera de sus etapas de desarrollo. [20]

#### **Aplicación de IDEF0**

- Como medio para comunicar reglas y procesos de negocios
- Para la obtención de la visión estratégica de cualquier proceso de negocio, educación, salud, etc.
- Facilitación del análisis para la identificación de áreas de mejora.

Se decidió utilizar esta técnica de modelado con el objetivo de facilitar la comunicación con el cliente respecto los procesos actuales de negocio, debido a que esta técnica permite realizar un modelado de procesos bien definidos con buena claridad y visibilidad al cliente, mediante actividades con sus entradas, salidas, controles y sujeto.

### 1.6.4. Lenguaje de Programación

Un lenguaje de programación es un idioma artificial dentro de la informática, nos permite crear programas mediante un conjunto de instrucciones, operadores y reglas de sintaxis que pone a disposición del programador para que este pueda comunicarse con los dispositivos hardware y software existentes.

#### Java

Java es un lenguaje de programación orientado a objetos. Fue desarrollado por la compañía Sun Microsystems a principios de los años 90. Una de sus principales características es la independencia de plataforma.

Se decidió utilizar Java como lenguaje ya que ha sido diseñado para producir software:

- Orientado a objetos: Beneficioso tanto para el proveedor de bibliotecas de clases como para el programador de aplicaciones.
- Robusto: Los errores se detectan en el momento de producirse, lo que facilita la depuración.
- Crear componentes de software y reusar software. Los programas de java consiste de piezas llamadas clases.[10]

### 1.6.5. Herramienta CASE

La tecnología CASE<sup>7</sup> supone la automatización del desarrollo del software, contribuyendo a mejorar la calidad y la productividad en el desarrollo de sistemas de información y se plantean los siguientes objetivos:

- Permitir la aplicación práctica de metodologías estructuradas, las cuales al ser realizadas con una herramienta se consigue agilizar el trabajo.
- Facilitar la realización de prototipos y el desarrollo conjunto de aplicaciones.
- Simplificar el mantenimiento de los programas.
- Mejorar y estandarizar la documentación.
- Facilitar la reutilización de componentes software.

---

<sup>7</sup>**CASE:** Por sus siglas en inglés *Computer Aided Software Engineering*. Diversas aplicaciones informáticas destinadas a aumentar la productividad en el desarrollo de software.

- Permitir un desarrollo y un refinamiento visual de las aplicaciones, mediante la utilización de gráficos. [21]

La mejor razón para la creación de estas herramientas fue el incremento en la velocidad de desarrollo de los sistemas. Por esto, las compañías pudieron desarrollar sistemas sin encarar el problema de tener cambios en las necesidades del negocio, antes de finalizar el proceso de desarrollo.[21]

### **Visual Paradigm 5.0**

Se ha decidido utilizar Visual Paradigm como herramienta CASE, debido fundamentalmente a que es multiplataforma y por ser una herramienta de diseño, que hace uso del Lenguaje Unificado de Modelado. Soporta el ciclo de vida completo del proceso de desarrollo del software: análisis, diseño, construcción, pruebas y despliegue a través de la representación de todo tipo de diagramas. Ofrece un diseño enfocado al negocio generando un software de mayor calidad, uso de un lenguaje estándar común para todo el equipo de desarrollo que facilita la comunicación y posee capacidades de ingeniería directa e inversa.

### **Embarcadero ER/Studio 8.0**

Herramienta líder para el modelado de datos, ayuda a las empresas a descubrir, documentar, y re-utilizar los activos de datos. Con soporte completo a las bases de datos, los arquitectos de las mismas tienen el poder de fácilmente realizar ingeniería a la inversa. Analizar y optimizar bases de datos existentes. La productividad gana y refuerza los estándares organizacionales pudiendo alcanzar la excelencia con las grandes capacidades colaborativas de ER/Studio.[22]

Provee una interfaz visual de fácil utilización para documentar, entender y publicar información acerca de las bases de datos existentes de tal forma que puedan ser mejor controladas para soportar los objetivos del negocio. Con poderosa ingeniería a la inversa de las principales bases de datos de la industria, permite al modelador de datos comparar y consolidar estructuras comunes de datos sin la necesidad de crear duplicaciones innecesarias.[22]

### **1.6.6. Framework de Desarrollo**

En el desarrollo de software, un **framework**, es una estructura de soporte definida en la cual otro proyecto de software puede ser organizado y desarrollado. Es una estructura de software compuesta por componentes personalizables e intercambiables para el desarrollo de una aplicación.[23]

Java soporta un gran número de framework, cada uno de ellos se especializa en una parte distinta del negocio, ya sea la capa de presentación, la capa de la lógica del negocio o la capa de acceso a datos.

Para el desarrollo del sistema se decidieron utilizar los siguientes framework: Java Server Faces (JSF) para la capa de presentación, Spring framework para la capa de negocio e Hibernate para la capa de acceso a datos.

### **Framework de Presentación. JSF**

JSF (Java Server Faces) es un framework que está basado en el conocido lenguaje de programación Java. JSF tiene como fin, simplificar el desarrollo de interfaces web en las aplicaciones JEE (Java Enterprise Edition).[24]

Como características principales de JSF podemos mencionar que utiliza el patrón MVC (Modelo Vista Controlador) en la creación de las aplicaciones de esta forma:

- JSF Utiliza páginas JSP para generar las vistas de usuario.
- Asocia cada vista que se crea a un conjunto de objetos java (clases) las cuales son conocidos como ManagedBeans estos facilitan la forma en que se manipula la información.

### **Framework para la lógica del negocio. Spring 2.5**

Spring es un framework de código abierto. La arquitectura de Spring está basada en el patrón de inyección de dependencias y puede trabajar de manera independiente o con servidores de aplicaciones existentes. Por otro lado hace uso de sus potentes archivos de configuración en XML<sup>8</sup>. Algunas de sus características que han propiciado esta decisión se basan en la enorme flexibilidad, frente a otras plataformas. Mientras que la integración del código de servicio en Spring se muestra como parte de la interfaz de programación, los desarrolladores de la aplicación tienen la flexibilidad de ensamblar los servicios que necesiten. [25]

Se decidió utilizar Spring para la lógica de negocio ya que entre sus principales características se encuentran:

- Es código abierto.
- Enfoque en el manejo de objetos de negocio, dentro de una arquitectura en capas.

---

<sup>8</sup>**XML**: por sus siglas en inglés *eXtensible Markup Language* (lenguaje de marcas extensible).

Una ventaja de Spring es su modularidad, pudiendo usar algunos de los módulos sin comprometerse con el uso del resto.

### Framework de acceso a datos. Hibernate 3

Hibernate, es el ORM<sup>9</sup> para la tecnología Java más utilizado, tiene una comunidad de desarrollo amplia. Permite el mapeo entre los campos de las tablas de la Base de Datos Relacional y los atributos de las Clases Persistentes de la aplicación, tanto mediante archivos de configuración XML como anotaciones.[26]

Hibernate posee un lenguaje propio llamado HQL para realizar consultas a la base de datos, lo que implica que no haya que modificar el código si se decide cambiar de Sistema Gestor de Base de Datos, sólo se modificaría un parámetro en el fichero de configuración de Hibernate.

Se decidió utilizar Hibernate por ser un poderoso medio para el mapeo de clases de Java a tablas de la base de datos. Además de ser Open Source y la licencia del producto está eximida en costo y posee un lenguaje propio.

### 1.6.7. Entorno de Desarrollo Integrado

“Un entorno de desarrollo integrado es un programa compuesto por un conjunto de herramientas útiles para un desarrollador de software”.[27]

El Entorno de Desarrollo (IDE) seleccionado es **Eclipse Indigo 3.7.2**. Eclipse es un IDE multiplataforma, libre para crear aplicaciones de cualquier tipo. Fue creado originalmente por International Business Machines Corporation (IBM), actualmente es desarrollado por la Fundación Eclipse, una organización que fomenta una comunidad de código abierto y un conjunto de productos complementarios, capacidades y servicios. Sirve como IDE Java y cuenta con numerosas herramientas de desarrollo de software. Entre sus principales ventajas se encuentran:

- Abierto y extensible.
- Compilación en tiempo real.
- Es multiplataforma.

---

<sup>9</sup> El **mapeo objeto-relacional** (más conocido por su nombre en inglés, *Object-Relational mapping*, o sus siglas O/RM, ORM, y O/R mapping) es una técnica de programación para convertir datos entre el sistema de tipos utilizado en un lenguaje de programación orientado a objetos y el utilizado en una base de datos relacional.

### 1.6.8. Sistema Gestor de Base de Datos

Un Sistema Gestor de Bases de Datos (SGBD) es una colección de programas cuyo objetivo es servir de interfaz entre la base de datos, el usuario y las aplicaciones. Se compone de un lenguaje de definición de datos, de un lenguaje de manipulación de datos y de un lenguaje de consulta. Un SGBD permiten definir los datos a distintos niveles de abstracción y manipular dichos datos, garantizando la seguridad e integridad de los mismos.[28]

#### PostgreSQL 8.2

PostgreSQL es un potente sistema de gestión de bases de datos objeto-relacional multiusuario, centralizado y de propósito general, que está siendo desarrollado desde 1977 y que está liberado bajo la licencia Berkeley Software Distribution (BSD). Fue pionero en muchos conceptos que estuvieron disponibles en algunos sistemas de bases de datos comerciales de alto calibre como por ejemplo: control de acceso simultáneo, gestión de transacciones, puntos de seguridad; fue uno de los primeros intentos en implementar un motor de bases de datos relacional.[29]

Se decidió utilizar como SGBD a **PostgreSQL 8.2** debido a que está diseñado con una arquitectura cliente/servidor y usa múltiples procesos para garantizar la estabilidad del sistema. Un fallo en uno de los procesos no afectará el resto y el sistema continuará funcionando. El código fuente se encuentra disponible para todos sin costo alguno. Debido a la liberación de la licencia, PostgreSQL se puede usar, modificar y distribuir de forma gratuita para cualquier fin, ya sea privado, comercial o académico.

### 1.6.9. Servidor Web

Un **servidor web** es un programa que gestiona una aplicación en el lado del servidor realizando conexiones, para enviar contenido a un cliente web que lo solicite. Los servidores web establecen conexiones bidireccionales y/o unidireccionales y síncronas o asíncronas con el cliente, enviando una respuesta en diferentes lenguajes. El código recibido por el cliente suele ser compilado y ejecutado por un Navegador Web. Para la transmisión de estos datos se utiliza algún protocolo, generalmente se utiliza el HTTP para estas comunicaciones. El término también se emplea para referirse al ordenador que ejecuta el programa.[30]

## **Servidor Web Apache Tomcat 6.0.26**

El servidor web apache es un servidor robusto, de código libre y modular, que facilita los servicios con los protocolos TCP/IP[30].

Es, según numerosos estudios, el principal servidor de la Web desde hace diez años. Respaldo por una comunidad de desarrollo brillante, se sustenta en un amplio número de personas y organizaciones, desde gigantes de la talla de IBM<sup>10</sup> hasta consultores particulares. Transparencia y diversidad son las principales características de Apache. El código fuente es totalmente abierto. Su arquitectura modular, construida sobre un pequeño núcleo, se adapta a las necesidades específicas de cada usuario.[31]

Para el desarrollo del sistema se utilizará el servidor web **Apache Tomcat 6.0.26** ya que es una implementación de las tecnologías Java Servlet y Java Server Pages, por esta razón, funciona en cualquier sistema operativo que disponga de la máquina virtual de Java. Se desarrolla en un entorno abierto y participativo, publicado bajo la licencia del software Apache. Se usa fundamentalmente en entornos con alto nivel de tráfico y alta disponibilidad.[32]

### **1.6.10. Sistema de Control de Versiones**

Un **Sistema de Control de Versiones** es un software que organiza y controla las distintas revisiones que se realicen sobre los elementos de un producto en su desarrollo.[33]

**Subversion** es un sistema centralizado para compartir información que fue diseñado como reemplazo de CVS (Concurrent Versions System). La parte principal de Subversion es el repositorio, el cual es un almacén central de datos. El repositorio guarda información en forma de árbol de archivos. Un número indeterminado de clientes puede conectarse al repositorio para leer o escribir en esos archivos.[33]

Ventajas:

- Las modificaciones (incluyendo cambios a varios archivos) son atómicas.
- La creación de ramas y etiquetas es una operación más eficiente.
- Se envían sólo las diferencias en ambas direcciones (en CVS siempre se envían al servidor archivos completos).

---

<sup>10</sup>IBM: por sus siglas en inglés *International Business Machines*.

Las razones fundamentales para el uso de esta herramienta en el desarrollo de la aplicación son: gestiona las modificaciones durante el desarrollo del sistema y permite que varias personas trabajen sobre los mismos ficheros.

### **1.6.11. Conclusiones parciales**

Después de realizar el análisis de los sistemas existentes, carentes de las funcionalidades necesarias para el desarrollo de los procesos aquí mencionados, se deriva la necesidad de una aplicación para el Centro de Telemática de la facultad 2, capaz de cumplir con las expectativas deseadas. Para ello se cuenta con el empleo de las herramientas aquí argumentadas y seleccionadas, las cuales tienen como punto en común, la característica de ser código abierto y poseer una amplia documentación por lo que podrán ser usadas para el logro del producto que se necesita.



### CAPÍTULO 2: CARACTERÍSTICAS DEL SISTEMA. EXPLORACIÓN Y PLANIFICACIÓN

#### Introducción

En el presente capítulo se mencionan los procesos que se desean automatizar, se describe la propuesta de solución. Se modela el negocio mediante la técnica IDEF0. Se identifican las personas relacionadas con el sistema. Se describen las historias de usuario que permitirán planificar las iteraciones en que se desarrollará el sistema además del plan de entrega de versiones especificándose la fecha en las que se entregan las versiones del sistema al cliente.

#### 2.1. Situación Problemática

El Centro de Telemática tiene como misión desarrollar sistemas y servicios informáticos en las ramas de las telecomunicaciones y la seguridad informática, altamente comprometidos con la Revolución, capaz de integrar los procesos docentes, productivos e investigativos de alto nivel; contando con un personal especializado en dichas áreas.[34]

Actualmente el centro está estructurado de la siguiente forma:



Figura 1: Estructura del Centro de Telemática

Para asignar las tareas de cada trabajador se conforma el Plan de Trabajo anualmente. El director del centro es el encargado de elaborar y entregar en tiempo y forma los informes sobre las actividades del centro así como elaborar, asignar y controlar el Plan de Trabajo de los Sub Directores, Asesores y de los Jefes de Departamento. El Jefe de Departamento a su vez es el encargado de gestionar y evaluar el cumplimiento del Plan de Trabajo de los trabajadores, apoyándose en la estructura organizativa de la facultad y del departamento.

Los directivos del centro manejan grandes cantidades de información referente a los planes de sus trabajadores. Actualmente esta gestión se realiza de forma manual aumentando la posibilidad de cometer errores. Se desperdicia tiempo pues no se tiene la información centralizada, y los directivos para asignar una actividad determinada deben buscar entre todos sus trabajadores aquellos que estén capacitados y disponibles para realizar una tarea.

Algunas actividades se planifican para un período de tiempo muy largo y no existe forma de verificarlas o ver el avance de las mismas cuando algún directivo del centro o de la Universidad lo solicita.

Existen tareas que no son lo más específicas y detalladas posibles y no se mide el por ciento de avance de las mismas. Como resultado del estudio realizado se arroja que en el centro no se realiza un control estricto sobre el avance de las tareas planificadas haciéndose imposible aprovechar al máximo las potencialidades del Plan de Trabajo.

### **2.2. Modelo de proceso del negocio**

Se define el proceso de confección de los Planes de Trabajo de cada trabajador utilizando la metodología IDEF con la notación IDEF0, para lograr un mejor entendimiento por parte de los clientes.

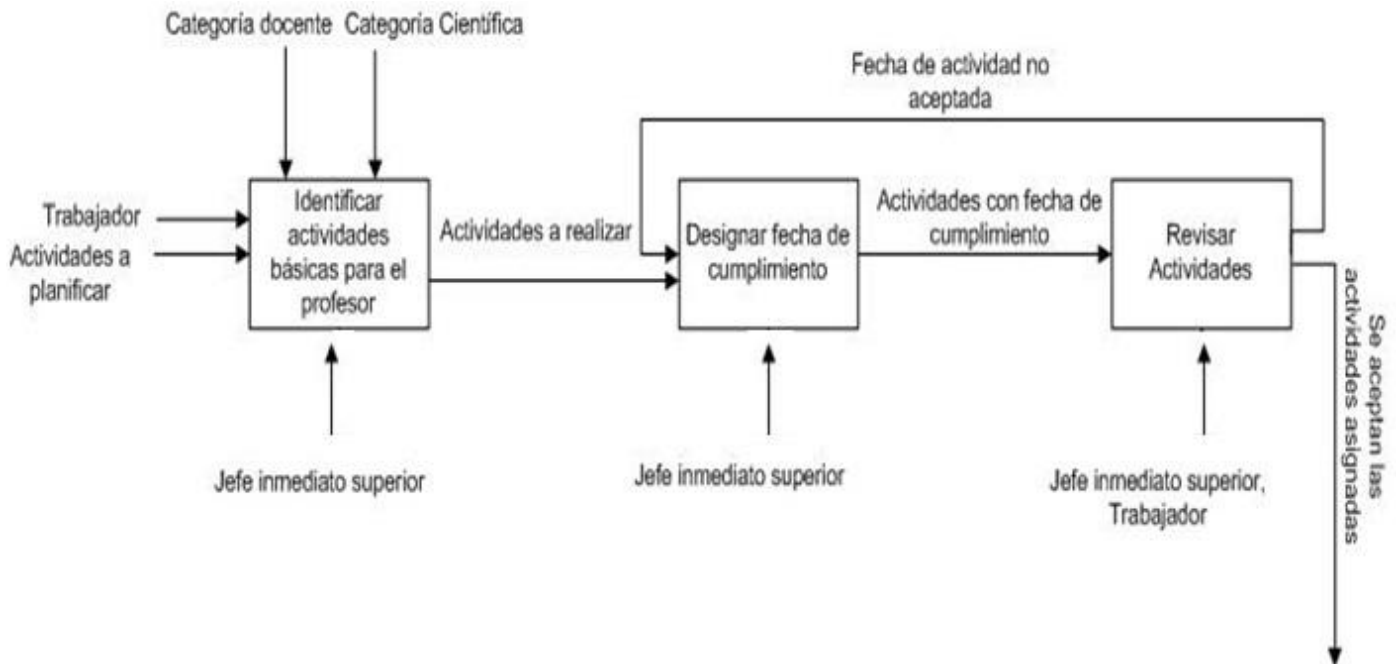


Figura 2: Modelo IDEF0. Proceso de Elaboración del Plan de Trabajo

El proceso comienza cuando los Directivos del centro identifican las actividades que conformarán el Plan de Trabajo. Luego se designan las fechas de cumplimiento para cada actividad. Seguido a esto el plan es revisado por el trabajador y su jefe inmediato superior. El plan solamente puede ser rechazado por causa de las fechas asignadas para el cumplimiento de las actividades ya que las tareas en él planteadas son de estricto cumplimiento, en caso de no aceptarse el plan propuesto por que las fechas asignadas afectan otras labores del trabajador entonces se modificará el plan hasta que ambas partes estén totalmente de acuerdo.

### 2.2.1. Identificar actividades básicas para el trabajador

La UCI anualmente define objetivos y estrategias que deben ser cumplidos por los trabajadores en el curso. Cada facultad recibe un conjunto de indicadores que deben cumplir para lograr con éxito los objetivos trazados por la Universidad. El decano de cada facultad es el encargo de informar a los jefes de centro y al consejo de dirección las actividades que deben cumplir. Los jefes de centro asignan las actividades y confeccionan el Plan de Trabajo de cada uno de los jefes de departamento quienes, a su

## Capítulo 2. Características del Sistema

---

vez, se encargan de gestionar los planes de cada uno de sus trabajadores. Para la confección del Plan de Trabajo se tienen en cuenta las áreas de resultados claves y sus actividades correspondientes:

- **Producción:** Los trabajadores pertenecientes a proyectos productivos desempeñarán tareas según el rol que ocupen en el mismo. Los roles en los proyectos productivos del Centro de Telemática son los siguientes: Gerente General, Líder de Proyecto, Arquitecto, Analista, Diseñador de Base de Datos, Programador, Gestor de la Configuración y Cambios y Gestor de la Calidad.
- **Ciencia y Técnica:** A nivel de Universidad se identifican los indicadores de ciencia y técnica para el año en curso, los mismos son desagregados por las diferentes facultades y a nivel de centro de desarrollo. Cada directivo del centro identifica la cantidad de eventos y publicaciones en los que debe participar el trabajador en el curso para cumplir con los indicadores del centro. Los mismos estarán detallados según su categoría docente y científica.
- **Postgrado:** Los trabajadores según su categoría docente deben impartir o recibir cursos de posgrado en el área temática en que ejercen funciones.
- **Formación del Profesional:** Los trabajadores que tengan plaza de profesor deben impartir asignaturas en pregrado. Deberán participar en los talleres metodológicos pertenecientes a las mismas. Solo podrán impartir clases los trabajadores que tengan categoría docente. En el caso de los recién graduados solo podrán impartir asignaturas optativas siempre y cuando el profesor principal de la asignatura tenga categoría docente. Para las tutorías de tesis no influye ningún tipo de categoría y el trabajador podrá desempeñarse como tutor, cotutor o consultor de tesis. Cada trabajador podrá formar parte de tribunales de tesis ya sea como presidente, vocal, secretario u oponente de la misma. Los profesores titulares y auxiliares deben dirigir procesos académicos, dirigir y participar en investigaciones científicas y en la formación científico educativo del personal con categoría científicas precedente.
- **Extensión Universitaria:** Cada trabajador debe cumplir estrictamente con la planificación que se realice de la guardia obrera, debe realizar dos visitas al mes a la beca de sus estudiantes, si posee vocación de baile, canto, actuación u otra manifestación cultural puede apoyar a su facultad en los festivales de artistas aficionados de igual manera podrá participar y apoyar los juegos deportivos que se realizan en la Universidad.

### **2.2.2. Designar fechas de cumplimiento:**

En el Plan de Trabajo deben quedar especificadas las fechas de cumplimiento de cada una de las actividades que lo conformen; las mismas no deberán coincidir para evitar afectar labores del trabajador o el incumplimiento en alguna de las tareas planificadas.

### **2.2.3. Revisar actividades:**

El Plan de Trabajo es aprobado por el directivo inmediato superior y el trabajador, donde se explicarán cada una de las actividades que conformarán el plan y sus respectivas fechas de cumplimiento. Estas actividades serán de estricto cumplimiento y solamente podrán ser rechazadas porque alguna actividad coincida con otra en las fechas de realización o el trabajador se vea afectado y presente una justificación.

### **2.3. Propuesta de solución**

Se implementará una aplicación web, capaz de gestionar todo el proceso de control y seguimiento de los planes de trabajo del Centro de Telemática. Dicha aplicación permitirá llevar de manera más organizada todos los datos e informaciones relacionadas con cada uno de los planes de los trabajadores, controlar el estado de cumplimiento de cada una de las tareas así como proponer al jefe de centro el personal que esté más calificado para cumplir con una tarea determinada. Además de enviarle una notificación a cada trabajador cuando se le asigne una actividad en la cual participe o sea el responsable. Se decide implementar una aplicación web para lograr que:

- Los usuarios puedan tener acceso a ella en el momento en que lo requieran con solamente tener instalado un navegador Web.
- Facilitar la actualización y mantenimiento, pues no se hace necesario distribuir e instalar el software en las PC de los usuarios.

Se tiene como propuesta de solución una aplicación web con la estructura que se muestra en la Figura 3.

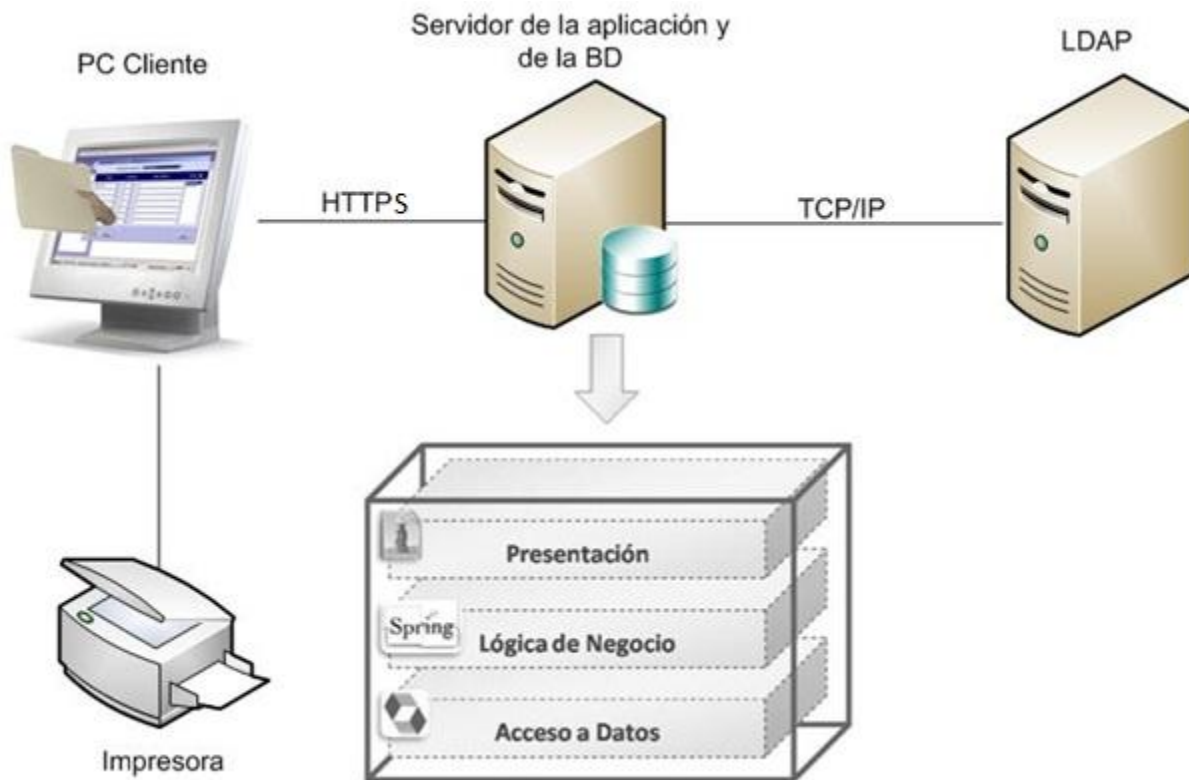


Figura 3: Propuesta de Solución.

### 2.4. Especificación de los requisitos del software

Los requisitos para un sistema de software determinan lo que hará el sistema y definen las restricciones de su operación e implementación. Los mismos se clasifican en funcionales y no funcionales. Los requisitos no funcionales son propiedades o cualidades que el producto debe tener. Debe pensarse en estas propiedades como las características que hacen al producto atractivo, usable, rápido o confiable.

#### 2.4.1. Requisitos no funcionales

##### Usabilidad:

**RnF1.** El sistema deberá contar con una interfaz de fácil entendimiento para que usuarios inexpertos puedan interactuar fácilmente con el software.

**RnF2.** El sistema podrá ser utilizado por cualquier usuario con las siguientes características:

- ✓ Conocimientos básicos relativos al uso de una computadora.
- ✓ Conocimientos básicos del sistema operativo Windows.

- ✓ Conocimientos sólidos relativos a los procesos de negocio acorde al rol que desempeñe.

**RnF3.** El sistema se distribuirá en lenguaje español.

### **Fiabilidad:**

**RnF4.** El sistema estará disponible las 24 horas del día y durante los 7 días de la semana, siempre que no existan fallos en la red.

**RnF5.** No se realizarán mantenimientos preventivos en horario laboral, deberán ejecutarse en un horario estipulado o los fines de semana, para no afectar la disponibilidad del sistema.

**RnF6.** Sólo se accederá a la BD desde la aplicación, nunca directamente desde el gestor de BD.

### **Rendimiento:**

**RnF7.** Los tiempos de respuesta del sistema estarán entre 2.05 y 2.30 segundos.

### **Soporte:**

**RnF8.** Debe poseer documentos técnicos como el Manual de Usuario.

### **Seguridad:**

**RnF10.** Confidencialidad: Los usuarios se autentican para poder acceder al sistema según los permisos asignados.

**RnF11.** Integridad: Se garantiza la integridad de la información que se maneja en el sistema ya que solo podrá ser modificada por las personas autorizadas.

**RnF12.** Disponibilidad: El sistema deberá estar disponible las 24 horas del día.

### **Interfaz:**

**RnF13.** Interfaz Web: La interfaz deberá ser sencilla con tonalidades azules claras y sin cúmulo de imágenes u objetos que distraigan al cliente del objetivo de su empleo.

**RnF14.** Las ventanas del sistema contendrán claro y bien estructurados los datos, y al mismo tiempo permitirán la interpretación correcta e inequívoca de la información.

### **Requisitos de hardware:**

**RnF15.** Para explotación del cliente: PC Pentium 3 o superior, CPU 133 MHZ o superior, 256 RAM mínimo 512 RAM recomendada o superior.

**RnF16.** Para explotación del servidor: CPU Dual Core 2.0 GHZ o superior, memoria RAM de 2 GB, 100 GB HDD.

### Requisitos de software:

**RnF17.** En el servidor sobre el cual se instalará la aplicación debe contar con un SO Windows o GNU/Linux.

**RnF18.** El servidor deberá contar además con la instalación de Gestor de Bases de Datos PostgreSQL y el servidor web ApacheTomcat.

### 2.5. Personas relacionadas con el Sistema

Persona encargada	Descripción
Administrador del Sistema	Es el encargado de asignarles los permisos a los usuarios en el sistema y de gestionar usuarios del sistema.
Controlador	Es el encargado de asignar y controlar las actividades de sus subordinados.
Supervisor	Es el encargado de supervisar todos los Planes de Trabajo pero no posee permisos para modificarlos.
Trabajador	Solamente tendrá permiso para ver su Plan de Trabajo, modificar su cuenta de usuario y actualizar el porcentaje de las actividades.

Tabla 1. Descripción de las personas relacionadas con el Sistema

### 2.6. Fase de Exploración

El ciclo de vida de un proyecto realizado con la metodología XP se inicia con la fase de Exploración. En esta fase, los clientes plantean a grandes rasgos las Historias de Usuarios. Al finalizar el equipo cuenta con suficiente material de trabajo como para producir una primera entrega. Al mismo tiempo el equipo de desarrollo se familiariza con las herramientas y tecnologías que serán utilizadas en el proyecto.

#### 2.6.1. Historias de Usuario

Las Historias de Usuario (HU) corresponden a la técnica utilizada para especificar los requisitos del software. Se trata de formatos en los cuales el cliente describe brevemente las características que el sistema debe poseer, sean requisitos funcionales o no funcionales. El tratamiento de las historias de



usuario es muy dinámico y flexible. Cada HU es lo suficientemente comprensible y delimitada para que los programadores puedan implementarla en unas semanas.[35]

A continuación se describen las HU determinadas por el cliente:

Historia de Usuario	
<b>Número:</b> 1	Nombre de Historia de Usuario: Autenticar Usuario
<b>Modificación de historia de Usuario</b>	
<b>Usuario:</b> Trabajador	<b>Iteración Asignada:</b> 1
<b>Prioridad en negocio:</b> Alta	
<b>Riesgo en Desarrollo:</b> Bajo	
<b>Descripción:</b> Se brinda la posibilidad de que toda persona que acceda al sistema introduzca sus datos (usuario y contraseña) con la finalidad de verificarle y otorgarle los permisos según el rol que ocupa dentro de la aplicación. En caso de que lo datos sean correctos se le permite el acceso, en caso contrario se le muestra un mensaje de aviso indicándole que los datos introducidos son incorrectos.	

Tabla 2.Descripción de la HU Autenticar Usuario

Historia de Usuario	
<b>Número:</b> 2	Nombre de Historia de Usuario: Gestionar Usuario Local
<b>Modificación de historia de Usuario:</b>	
<b>Usuario:</b> Administrador	<b>Iteración Asignada:</b> 1
<b>Prioridad en negocio:</b> Alta	
<b>Riesgo en Desarrollo:</b> Bajo	
<b>Descripción:</b> Se brinda la posibilidad de insertar, modificar, listar y eliminar usuarios en el sistema. Se incluyen los datos del usuario: <ul style="list-style-type: none"> <li>● Usuario</li> <li>● Contraseña</li> </ul>	

Tabla 3.Descripción de la HU Gestionar Usuario Local

Historia de Usuario	
<b>Número:</b> 3	Nombre de Historia de Usuario: Gestionar Trabajador
<b>Modificación de historia de Usuario:</b>	

<b>Usuario:</b> Administrador	<b>Iteración Asignada:</b> 1
<b>Prioridad en negocio:</b> Alta	
<b>Riesgo en Desarrollo:</b> Bajo	
<p><b>Descripción:</b> Se brinda la posibilidad de insertar, modificar, listar o eliminar un trabajador en el sistema. Luego se mostrará un mensaje de confirmación de la acción realizada. Para insertar un trabajador el sistema mostrará los datos a llenar:</p> <ul style="list-style-type: none"> <li>• Nombre</li> <li>• Primer Apellido</li> <li>• Segundo apellido</li> <li>• Solapín</li> <li>• Facultad</li> <li>• Departamento</li> <li>• Categoría Docente</li> <li>• Categoría Científica</li> </ul> <p>Estas dos últimas no son de carácter obligatorio.</p>	

Tabla 4. Descripción de la HU Gestionar Trabajador

Historia de Usuario	
<b>Número:</b> 4	Nombre de Historia de Usuario: Gestionar Actividad
<b>Modificación de historia de Usuario:</b>	
<b>Usuario:</b> Controlador	<b>Iteración Asignada:</b> 1
<b>Prioridad en negocio:</b> Alta	
<b>Riesgo en Desarrollo:</b> Bajo	
<p><b>Descripción:</b> Se realiza la acción de insertar, modificar, listar o eliminar una actividad que no esté definida en el Plan de Trabajo anual. Se introducirán los datos de la actividad:</p> <ul style="list-style-type: none"> <li>• Nombre</li> <li>• Fecha de Inicio</li> <li>• Fecha Fin</li> <li>• Tipo de actividad</li> </ul>	

- Responsable

Tabla 5.Descripción de la HU Gestionar Actividad

Historia de Usuario	
<b>Número:5</b>	Nombre de Historia de Usuario: Asignar Actividad
<b>Modificación de historia de Usuario:</b>	
<b>Usuario:</b> Controlador	<b>Iteración Asignada:2</b>
<b>Prioridad en negocio:</b> Alta	
<b>Riesgo en Desarrollo:</b> Bajo	
<b>Descripción:</b> Se realiza la acción de asignar una actividad determinada a uno o varios trabajadores.	

Tabla 6.Descripcion de la HU Asignar Actividad

Historia de Usuario	
<b>Número:6</b>	Nombre de Historia de Usuario: Gestionar Asignatura
<b>Modificación de historia de Usuario:</b>	
<b>Usuario:</b> Controlador	<b>Iteración Asignada:3</b>
<b>Prioridad en negocio:</b> Baja	
<b>Riesgo en Desarrollo:</b> Bajo	
<b>Descripción:</b> Se brinda la posibilidad de insertar, modificar, listar y eliminar asignaturas que se impartirán. Se introducirán los siguientes atributos de la asignatura: <ul style="list-style-type: none"> <li>• Nombre</li> <li>• Semestre</li> <li>• Curso</li> <li>• Departamento</li> </ul> Los profesores solamente podrán impartir de 1 a 2 a signaturas por semestre.	

Tabla 7.Descripción de la HU Gestionar Asignatura

Historia de Usuario	
<b>Número:7</b>	Nombre de Historia de Usuario: Gestionar Curso de Superación
<b>Modificación de historia de Usuario:</b>	

<b>Usuario:</b> Controlador	<b>Iteración Asignada:</b> 3
<b>Prioridad en negocio:</b> Baja	
<b>Riesgo en Desarrollo:</b> Bajo	
<p><b>Descripción:</b> Se brinda la posibilidad de insertar, modificar y eliminar un Curso de Superación. Se introducirán los siguientes atributos del Curso de Superación:</p> <ul style="list-style-type: none"> <li>● Nombre</li> <li>● Fecha inicial</li> <li>● Fecha Final</li> <li>● Departamento</li> </ul>	

Tabla 8.Descripción de la HU Gestionar Curso de Superación

Historia de Usuario	
<b>Número:</b> 8	Nombre de Historia de Usuario: Gestionar Evento
<b>Modificación de historia de Usuario:</b>	
<b>Usuario:</b> Controlador	<b>Iteración Asignada:</b> 3
<b>Prioridad en negocio:</b> Baja	
<b>Riesgo en Desarrollo:</b> Bajo	
<p><b>Descripción:</b> Se brinda la posibilidad de insertar, modificar, listar y eliminar un Evento. Se introducirán los siguientes datos del evento:</p> <ul style="list-style-type: none"> <li>● Nombre</li> <li>● Fecha Inicial</li> <li>● Fecha Final</li> <li>● Tipo de evento: <ul style="list-style-type: none"> <li>▪ Científico</li> <li>▪ Extensionistas</li> <li>▪ Docente</li> <li>▪ Expositivo</li> <li>▪ Formativo</li> <li>▪ Investigativo</li> <li>▪ Productivo</li> </ul> </li> </ul>	

- Temática
- Descripción

Tabla 9. Descripción de la HU Gestionar Evento

Historia de Usuario	
Número:9	Nombre de Historia de Usuario: Proponer equipo de trabajadores para una actividad
<b>Modificación de historia de Usuario:</b>	
Usuario: Controlador	Iteración Asignada:2
Prioridad en negocio: Media	
Riesgo en Desarrollo: Bajo	
Descripción: Se brinda la posibilidad de proponer el equipo de trabajo más idóneo para efectuar una actividad determinada, utilizando para su realización técnicas de inteligencia artificial.	

Tabla 10. Descripción de la HU Proponer equipo de trabajadores para una actividad

Historia de Usuario	
Número:10	Nombre de Historia de Usuario: Proponer trabajador para una actividad
<b>Modificación de historia de Usuario:</b>	
Usuario: Controlador	Iteración Asignada:2
Prioridad en negocio: Media	
Riesgo en Desarrollo: Bajo	
Descripción: Se brinda la posibilidad de proponer el trabajador más idóneo para efectuar una actividad determinada, utilizando para su realización técnicas de inteligencia artificial.	

Tabla 11. Descripción de la HU Proponer trabajador para una actividad

Historia de Usuario	
Número:11	Nombre de Historia de Usuario: Enviar Notificación
<b>Modificación de historia de Usuario:</b>	
Usuario: Trabajador	Iteración Asignada:2
Prioridad en negocio: Media	
Riesgo en Desarrollo: Bajo	
Descripción: Se permitirá mandar una notificación al trabajador cuando se le asigne una actividad.	

## Capítulo 2. Características del Sistema

Tabla 12. Descripción de la HU Enviar Notificación

Historia de Usuario	
Número:12	Nombre de Historia de Usuario: Generar reporte de planillas de planes de trabajo
<b>Modificación de historia de Usuario:</b>	
Usuario: Trabajador	<b>Iteración Asignada:2</b>
<b>Prioridad en negocio:</b> Media	
<b>Riesgo en Desarrollo:</b> Bajo	
<b>Descripción:</b> Se permitirá exportar en formato PDF las planillas de los planes de trabajo de cada trabajador del centro.	

Tabla 13. Descripción de la HU Generar reporte de planillas de planes de trabajo

Historia de Usuario	
Número:13	Nombre de Historia de Usuario: Mostrar Plan de Trabajo.
<b>Modificación de historia de Usuario:</b>	
Usuario: Trabajador	<b>Iteración Asignada:1</b>
<b>Prioridad en negocio:</b> Alta	
<b>Riesgo en Desarrollo:</b> Bajo	
<b>Descripción:</b> Se permitirá mostrar el plan de trabajo de un trabajador determinado.	

Tabla 14. Descripción de la HU Mostrar Plan de trabajo

Historia de Usuario	
Número:14	Nombre de Historia de Usuario: Generar Reporte de Trabajadores
<b>Modificación de historia de Usuario:</b>	
Usuario: Controlador	<b>Iteración Asignada:2</b>
<b>Prioridad en negocio:</b> Media	
<b>Riesgo en Desarrollo:</b> Bajo	
<b>Descripción:</b> Se permitirá exportar en formato PDF un listado de trabajadores atendiendo a una determinada búsqueda.	

Tabla 15. Descripción de la HU Generar Reporte de Trabajadores

Historia de Usuario	
Número:15	Nombre de Historia de Usuario: Actualizar realización de actividad
<b>Modificación de historia de Usuario:</b>	
Usuario: Trabajador	Iteración Asignada:2
Prioridad en negocio: Media	
Riesgo en Desarrollo: Bajo	
Descripción: Se permitirá que el usuario actualice el porcentaje de realización de una tarea.	

Tabla 16. Descripción de la HU Actualizar realización de actividad

### 2.7. Fase de planificación.

En esta fase el cliente establece la prioridad que tendrá cada HU según sus necesidades más inmediatas, luego los programadores realizan una estimación del esfuerzo que se necesita para cada una de ellas. Además se toman acuerdos sobre el contenido de las entregas.

#### 2.7.1. Estimación de esfuerzo por HU.

Las estimaciones de esfuerzo asociado a la implementación de las HU las establecen los programadores, para ello utilizan puntos. Un punto, equivale a una semana ideal de programación.

Para el logro del sistema se ha realizado la estimación de esfuerzo por cada HU, teniendo en cuenta la complejidad de cada una de las funcionalidades requeridas por el cliente. A continuación se muestran las estimaciones de esfuerzo para cada HU.

Historia de Usuario	Estimación
Autenticar Usuario	0.5
Gestionar Usuario Local	0.4
Gestionar Trabajador	0.3
Gestionar Actividad	0.5
Asignar Actividad	0.1
Gestionar Asignaturas	0.2
Gestionar Cursos de Superación	0.2

Gestionar Evento	0.5
Proponer equipo de trabajadores para una actividad	1
Proponer trabajador para una actividad	1
Enviar Notificación	0.4
Generar reporte de planes de trabajo	0.2
Mostrar Plan de Trabajo	0.3
Generar reporte de trabajadores	0.2
Actualizar realización de actividad	0.1

Tabla 17. Estimación de esfuerzo por HU

### 2.7.2. Plan de iteraciones.

El plan de iteraciones muestra cuales son las HU que serán implementadas en cada iteración del sistema y las posibles fechas de liberaciones.

En la primera iteración se puede intentar establecer una arquitectura del sistema que pueda ser utilizada durante el transcurso del proyecto. Esto se logra escogiendo las historias que fueren la creación de esta arquitectura, sin embargo, esto no siempre es posible ya que es el cliente quien decide cuales son las historias que se implementarán en cada iteración. Al final de la última iteración el sistema estará listo para entrar en producción. Tratando de esta manera de tener preparadas las funcionalidades básicas e indispensables del sistema.

Se decide realizar el sistema en tres iteraciones las cuales se detallan a continuación:

**Iteración 1:** En esta iteración se implementan las Historias de Usuarios que se consideraron más necesarias atendiendo a su relevancia e impacto para el negocio, estas nos brindan la posibilidad de autenticarse cada usuario en el sistema, de gestionar los eventos, los usuarios locales, y los trabajadores. Tiene como finalidad administrar los valores iniciales para conformar el Plan de Trabajo. Como resultado de la iteración se tendrán implementadas las HU: 1, 2, 3, 5, 8

**Iteración 2:** En esta iteración se implementan las HU con prioridad media, estas nos brindan la posibilidad de gestionar actividades que no se encuentren en el Plan de Trabajo, gestionar las asignaturas que serán



impartidas por los profesores del centro, gestionar los cursos de superación, mostrar el Plan de Trabajo y generar reportes en formato PDF con el Plan de Trabajo y con el listado de los trabajadores de acuerdo a un criterio de búsqueda. Al finalizar se tendrán implementadas las HU: 4, 6, 7,12, 13, 14,15.

**Iteración 3:** En esta iteración se implementan las HU con prioridad baja. Al finalizar se tendrán implementadas las HU: 9, 10, 11. Estas funcionalidades brindan cierta comodidad en la gestión de las funcionalidades de prioridad alta. A partir de este momento quedará conformado totalmente el sistema y será sometido a pruebas.

### 2.7.3. Plan de duración de iteraciones.

Como parte del ciclo de vida de un proyecto utilizando la metodología XP se crea el plan de duración de iteraciones. Para la confección del plan se tiene en cuenta la estimación de esfuerzo para cada HU. Este plan permite mostrar la duración de cada iteración así como el orden en que se implementarán las HU, teniendo una mayor organización.

Iteraciones	Orden de las Historias de Usuario a implementar	Duración total de la iteración
1	<ol style="list-style-type: none"> <li>1. Autenticar Usuario.</li> <li>2. Gestionar Usuario Local.</li> <li>3. Gestionar Trabajador.</li> <li>4. Gestionar Evento.</li> <li>5. Asignar Actividad</li> </ol>	2 semanas y 4 días
2	<ol style="list-style-type: none"> <li>1. Gestionar Actividad.</li> <li>2. Gestionar Asignatura.</li> <li>3. Gestionar Curso de Superación.</li> <li>4. Generar reporte del Plan de Trabajo.</li> <li>5. Mostrar el Plan de Trabajo.</li> <li>6. Generar reporte de trabajadores.</li> <li>7. Actualizar realización de tarea.</li> </ol>	2 semanas y 4 día
3	<ol style="list-style-type: none"> <li>1. Proponer equipo de trabajadores para</li> </ol>	2 semana y 3 días

	una actividad. 2. Proponer trabajador para una actividad 3. Enviar Notificación.	
--	--	--

Tabla 18. Plan de Duración de Iteraciones

### 2.7.4. Plan de entregas

Con el plan de entrega se da un aproximado de las entregas de las versiones contando que la implementación comienza en el mes de marzo:

A entregar	Final 1ra iteración	Final 2da iteración	Final 3ra iteración
Sistema de seguimiento y control de los Planes de Trabajo del Centro de Telemática	31/3/2012 Versión 0.1	19/4/2012 Versión 0.2	6/5/2012 Versión 1.0

Tabla 19. Plan de entregas

### 2.8. Conclusiones Parciales

Durante el desarrollo de este capítulo se desarrolló la propuesta de solución que se desea implementar. Para esto se definieron los requisitos no funcionales y las HU propuestas por el cliente. Se realizó la estimación de esfuerzo necesaria para cada HU, y por último se construyeron el Plan de duración de las iteraciones y el Plan de entregas. La descripción de las HU será un aspecto fundamental en el desarrollo del sistema, permitiendo que los desarrolladores comiencen a implementar un producto que cumpla con toda la calidad requerida.

### CAPÍTULO 3: DISEÑO DEL SISTEMA

#### Introducción

Luego de las fases de Exploración y Planificación explicadas en el capítulo anterior, la metodología XP establece la fase de Diseño. Durante esta fase se comienza a definir el diseño arquitectónico del sistema partiendo de las HU. Se expondrán los patrones utilizados, así como el diseño de la Base de Datos.

#### 3.1. Diseño del Sistema

XP establece prácticas sobre el diseño para lograr un sistema robusto y reutilizable manteniendo su simplicidad. Logrando que cambios o modificaciones futuras puedan realizarse de manera más sencilla. XP no define una técnica específica de modelado, pueden utilizarse indistintamente sencillos esquemas, tarjetas CRC (Clase, Responsabilidad, Colaboración) o diagramas de clase utilizando UML, siempre que sean útiles y no requieran mucho tiempo en su creación.

Durante el diseño se aplican los patrones **GRASP** (General Responsibility Assignment Software Patterns) para la asignación de responsabilidades a las clases, además del estilo arquitectónico N capas, con el propósito de separar en componentes diferentes los datos de la aplicación, la interfaz de usuario y la lógica de negocio.

##### 3.1.1. Patrones Aplicados

#### Patrones de diseño.

Los patrones de diseño son descripciones de clases y objetos relacionados que están particularizados para resolver un problema de diseño general en determinado contexto.[36]

Un patrón de diseño nomina, abstrae e identifica los aspectos clave de una estructura de diseño común, lo que los hace útiles para crear un diseño orientado a objetos reutilizable. El patrón de diseño identifica las clases e instancias participantes, sus roles y colaboraciones y la distribución de responsabilidades.[36]

Los patrones **GRASP** son patrones generales de software para la asignación de responsabilidades a objetos. Describen los principios fundamentales del diseño de objetos para la asignación de responsabilidades.

- **Controlador:** Es utilizado como intermediario entre las interfaces y el algoritmo que las implementa. Este patrón sugiere que la lógica del negocio este separada de la capa de

presentación ya que esto posibilita una mayor reutilización del código y un mayor control. En la aplicación este patrón se evidencia en la creación de un BeanManejado para cada clase entidad.



- **Experto:** Este patrón nos indica que la implementación de un determinado método o la creación de un objeto debe recaer sobre la clase que conoce toda la información para crearlo. En la aplicación se evidencia este patrón en la clase AdministrarEventosManejado.java en el segmento de código que se muestra a continuación:

```
public void incluir(ActionEvent e) {
    try {
        Date hoy = new Date();
        if(nombre!="" && tipoEvento!=null && fechaInicio!=null
        && fechaFin!=null && hoy.before(fechaInicio) && fechaInicio.before(fechaFin))
        {
            Evento evento = new Evento(fechaInicio,
            fechaFin, nombre, tipoEvento, tematica, descripcion);
            getAdministracionService().incluirEvento(evento);
            reset();
        }
    }
    else {
        throw new Exception();
    }
} catch (Exception ex) {
    mensajeInformacion("Error al incluir el Evento, verifique sus
datos.");
}
}
```

- **Alta Cohesión:** Este patrón es utilizado para evita asignar demasiadas responsabilidades a las clases. Brinda la posibilidad de que la información que se almacene en cada clase sea coherente. Este patrón se evidencia en el sistema con la existencia de un BeanManejado que se encarga de manejar la lógica del negocio de la entidad que controla.

- **Bajo Acoplamiento:** Este patrón es el encargado de disminuir la dependencia entre clases, posibilitando que la modificación en alguna de ellas repercuta lo menos posible en las demás clases.

### 3.1.2. Estilo arquitectónico

La Arquitectura de Software es la organización fundamental de un sistema encarnada en sus componentes, las relaciones entre ellos y el ambiente y los principios que orientan su diseño y evolución.[37]

La arquitectura del software guía al equipo de desarrollo a través del ciclo de vida del sistema. Mediante los estilos y patrones arquitectónicos se logra representar el sistema de forma estructurada y organizada. Un estilo es un concepto descriptivo que define una forma de articulación u organización arquitectónica.[37]

#### **Arquitectura en capas**

El estilo en capas proporciona una organización jerárquica en la que cada capa proporciona servicios a la capa inmediata superior y se sirve de las prestaciones que le brinda la inmediata inferior.

El estilo arquitectónico para la implementación del Sistema de Seguimiento y Control de los Planes de Trabajo del Centro de Telemática es el N capas (tres capas), quedando estructurado en las capas de Presentación, Negocio y Acceso a Datos.

**Capa de Presentación:** Es la capa que interactúa con el usuario y la encargada de presentar y recolectar la información necesaria para hacer uso de los servicios de la Capa de Negocio, el framework que opera en esta capa es Java Server Faces.

**Capa del Negocio:** En esta capa están contenidas las clases del dominio de la aplicación, sus relaciones y las reglas de negocios que cumplen. Además, es la encargada de brindarle servicios a la Capa de Presentación, el framework que opera en esta capa es Spring.

**Capa de Acceso a Datos:** Ocupa el nivel más bajo en la jerarquía de la arquitectura en tres capas, en esta capa está contenida la lógica que lleva o trae información entre la Capa de Negocio y la fuente de dato. Como framework de acceso a la Base de Datos se utiliza Hibernate.

Esta arquitectura logra mayor aislamiento y separación entre las capas, lo que ayuda a eliminar notablemente la fragilidad del sistema ante cambios, además de permitir la reutilización de código. La capa de Presentación se comunica solamente con la capa de Negocio y la capa de Negocio se comunica con la capa de Acceso a Datos.

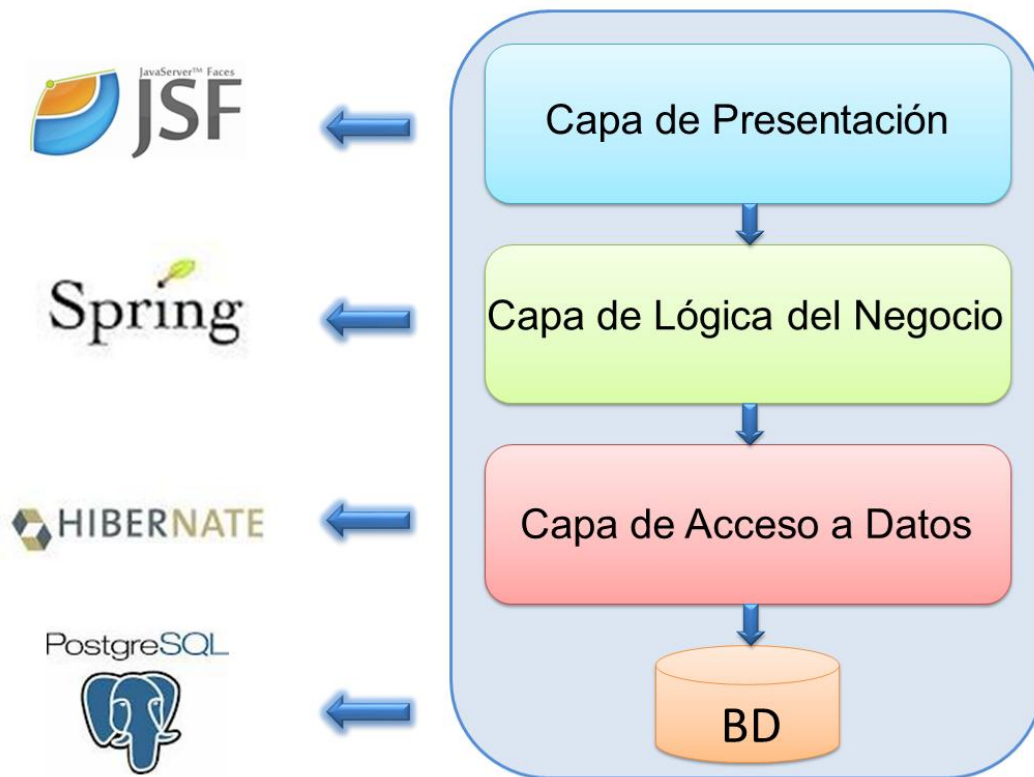


Figura 4: Representación del Patrón N-Capas

### 3.1.3. Tarjetas CRC (Clase-Responsabilidad-Colaboración)

La metodología XP no obliga a la realización de diagramas UML para representar las clases que contendrán la lógica del negocio, en su lugar propone el uso de tarjetas CRC (Clase-Responsabilidad-Colaboración) como técnica de modelado para ayudar a los desarrolladores de software a crear diseños de clases orientados a responsabilidades. Las tarjetas CRC permiten trabajar con una metodología orientada a objetos, cada tarjeta representa un objeto.

El título de la tarjeta corresponde al nombre de la clase, las Responsabilidades son colocadas en la izquierda y las Colaboraciones en la parte derecha como se muestra a continuación:

Actividad	
Responsabilidad	Colaboraciones
Contiene y manipula los siguientes atributos que hacen persistir los datos en memoria, mientras esta se encuentra en ejecución: <b>String</b> nombre <b>Date</b> fechaInicio <b>Date</b> fechaFin	Evento Asignatura Otra Actividad Proyecto AsignacionEvento AsignacionAsignatura AsignacionOtraActividad

Tabla 20: Tarjeta CRC clase Actividad

Asignación	
Responsabilidad	Colaboraciones
Contiene y manipula los siguientes atributos que hacen persistir los datos en memoria, mientras esta se encuentra en ejecución: <b>Trabajador</b> trabajador <b>Date</b> fechaAsignacion	Trabajador

Tabla 21: Tarjeta CRC clase Asignacion

Evento	
Responsabilidad	Colaboraciones
Contiene y manipula los siguientes atributos que hacen persistir los datos en memoria, mientras esta se encuentra en ejecución: <b>NTEvento</b> tipoEvento	Actividad NTEvento NRolEvento

Tabla 22: Tarjeta CRC clase Evento

Asignatura	
Responsabilidad	Colaboraciones

<p>Contiene y manipula los siguientes atributos que hacen persistir los datos en memoria, mientras esta se encuentra en ejecución:</p> <p><b>String</b> nombre</p>	<p>Actividad</p>
--	------------------

Tabla 23: Tarjeta CRC clase Asignatura

Proyecto	
Responsabilidad	Colaboraciones
<p>Contiene y manipula los siguientes atributos que hacen persistir los datos en memoria, mientras esta se encuentra en ejecución:</p> <p><b>String</b> nombre</p>	<p>Actividad</p>

Tabla 24: Tarjeta CRC clase Proyecto

OtraActividad	
Responsabilidad	Colaboraciones
<p>Contiene y manipula los siguientes atributos que hacen persistir los datos en memoria, mientras esta se encuentra en ejecución:</p> <p><b>String</b> nombre</p>	<p>Actividad</p>

Tabla 25: Tarjeta CRC clase OtraActividad

Trabajador	
Responsabilidad	Colaboraciones
<p>Contiene y manipula los siguientes atributos que hacen persistir los datos en memoria, mientras esta se encuentra en ejecución:</p> <p><b>String</b> nombre</p> <p><b>String</b> apellido1</p> <p><b>String</b> apellido2</p> <p><b>String</b> solapin</p>	<p>Asignacion</p>



<b>String</b> facultad <b>String</b> departamento	
--	--

Tabla 26: Tarjeta CRC clase Trabajador

### 3.1.4. Prototipos de interfaz de usuario

Un prototipo es una representación o demostración de un sistema. Los prototipos reducen el riesgo de construir productos que no satisfagan las necesidades de los clientes ya que si al usuario no le gusta una parte del prototipo este problema se corrige hasta que el cliente quede satisfecho.

El prototipo de interfaz de usuario (IU) permite comprobar cuestiones tales como la navegación, visualización de los atributos definidos así como la ejecución de todos los servicios.

A continuación se muestran los prototipos de interfaces de usuarios correspondientes a las HU de la primera iteración, los restantes prototipos de interfaces se muestran en el Anexo #5.

SC  
Seguimiento y Control

INICIO DE SESIÓN:

Nombre de Usuario

Contraseña

Dominio UCI  Sí  No

IU: Autenticar Usuario



## Sistema de Seguimiento y Control de los Planes de Trabajo

Conectado como: administrador (Desconexión)

- ▼ **Seguridad**
  - Incluir Usuario
  - Buscar Usuarios
- ▼ **Administración**
  - Gestionar Eventos
  - Gestionar Asignaturas
  - Gestionar Cursos de Superación
  - Gestionar Tesis
  - Gestionar Actividades
  - Gestionar Trabajador
  - Sugerir Trabajador
  - Adicionar Nomencladores
- ▼ **Soporte**
  - Manual de Usuario

### ADMINISTRAR EVENTOS

#### Datos del Nuevo Evento

Nombre	Fecha Inicial	Fecha Final	Tipo de Evento
<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
Temática	Descripción		
<input type="text"/>	<input type="text"/>		
<input type="button" value="Incluir"/>			<input type="button" value="Cerrar"/>

#### Eventos Registrados

Nombre	Desde	Hasta	Tipo de Evento	Modificar	Eliminar
JC	2012-04-03	2012-04-11	Científico	Modificar	Eliminar
Mi WebxCuba	2012-05-22	2012-05-25	Expositivo	Modificar	Eliminar
Tutoria de Tesis de Diploma	2012-05-25	2012-06-21	Docente	Modificar	Eliminar
Fordes	2012-04-19	2012-04-30	Extensionista	Modificar	Eliminar
Prueba	2012-05-30	2012-05-31	Docente	Modificar	Eliminar
CCIA	2012-10-10	2012-10-12	Científico	Modificar	Eliminar
Ultimo	2012-06-01	2012-06-08	Docente	Modificar	Eliminar
Graduacion	2012-06-02	2012-06-06	Docente	Modificar	Eliminar

IU: Gestionar Evento



# Sistema de Seguimiento y Control de los Planes de Trabajo

Conectado como: administrador (Desconexión)

- ▼ Seguridad
  - Incluir Usuario
  - Buscar Usuarios
- ▼ Administración
  - Gestionar Eventos
  - Gestionar Asignaturas
  - Gestionar Cursos de Superación
  - Gestionar Tesis
  - Gestionar Actividades
  - Gestionar Trabajador
  - Sugerir Trabajador
  - Adicionar Nomencladores
- ▼ Soporte
  - Manual de Usuario

## CONSULTAR TRABAJADOR

### Criterios de Búsqueda

#### Trabajador:

Nombre  1er Apellido  2do Apellido  Solapín   
 Facultad  Departamento  Categoría Docente  Categoría Científica

### Listado de Resultados

Nombre y Apellidos	Mostrar Plan	Asignar Actividades	Modificar	Eliminar
Bárbara Triana Morales	Mostrar	Asignar Actividades	Modificar	Eliminar
Maité Benitez Ortiz	Mostrar	Asignar Actividades	Modificar	Eliminar
José Alberto Betancourt	Mostrar	Asignar Actividades	Modificar	Eliminar
Denys Buedo Hidalgo	Mostrar	Asignar Actividades	Modificar	Eliminar
David Rodr Rod	Mostrar	Asignar Actividades	Modificar	Eliminar
Alexei Sanchez Figueroa	Mostrar	Asignar Actividades	Modificar	Eliminar
Yamila Izquierdo Vazquez	Mostrar	Asignar Actividades	Modificar	Eliminar

IU: Gestionar Trabajador



Seguridad

- Incluir Usuario
- Buscar Usuarios

Administración

- Gestionar Eventos
- Gestionar Asignaturas
- Gestionar Cursos de Superación
- Gestionar Tesis
- Gestionar Actividades
- Gestionar Trabajador
- Sugerir Trabajador
- Adicionar Nomencladores

Soporte

- Manual de Usuario

ASIGNAR ACTIVIDADES

Trabajador:

Nombre:	Bárbara	Facultad:	Facultad #2
Primer Apellido:	Triana	Solapín:	T15751
Segundo Apellido:	Morales	Departamento:	Consejo de Dirección

Filtrar Actividades:

Desde:  Hasta:  Tipo de Actividad:  Asignada:

Listado de Resultados



Actividad	Fecha Inicial	Fecha Final	Tipo	Asignación
Matemática 1	2012-09-15	2013-02-15	Asignatura	No Asignada
Matemática 2	2013-02-22	2013-06-15	Asignatura	Asignada
Curso de Inglés	2013-02-22	2013-06-15	Asignatura	No Asignada
BD	2012-09-15	2013-02-15	Asignatura	No Asignada
Programación 2	2012-09-15	2013-02-15	Asignatura	No Asignada
CCIA	2012-10-10	2012-10-12	Evento	No Asignada
Inteligencia Artificial	2012-09-15	2013-02-15	Asignatura	No Asignada
Seguridad Informática	2012-09-15	2013-02-15	Asignatura	No Asignada

IU: Asignar Actividad

SC Seguimiento y Control

# Sistema de Seguimiento y Control de los Planes de Trabajo

Conectado como: administrador (Desconexión)

- ▼ Seguridad
  - Incluir Usuario
  - Buscar Usuarios
- ▼ Administración
  - Gestionar Eventos
  - Gestionar Asignaturas
  - Gestionar Cursos de Superación
  - Gestionar Tesis
  - Gestionar Actividades
  - Gestionar Trabajador
  - Sugerir Trabajador
  - Adicionar Nomencladores
- ▼ Soporte
  - Manual de Usuario

### INCLUIR USUARIO

**Datos del Usuario**

Datos del usuario fuera del dominio UCI

Nombre de Usuario  Contraseña

Incluir Cancelar

S&C ©Copyright 2011-2012 Facultad 2

IU: Incluir Usuario Local

### 3.1.5. Diseño de la Base de Datos

El diseño de la Base de Datos se realiza con el objetivo de garantizar que los datos persistentes se almacenen de forma coherente. A continuación se muestra una parte del diseño de la Base de Datos definido para el Sistema de Seguimiento y Control de los Planes de Trabajo donde se muestran las entidades más importantes para el negocio. Para consultar el diseño de la Base de Datos completo ver Anexo 4.

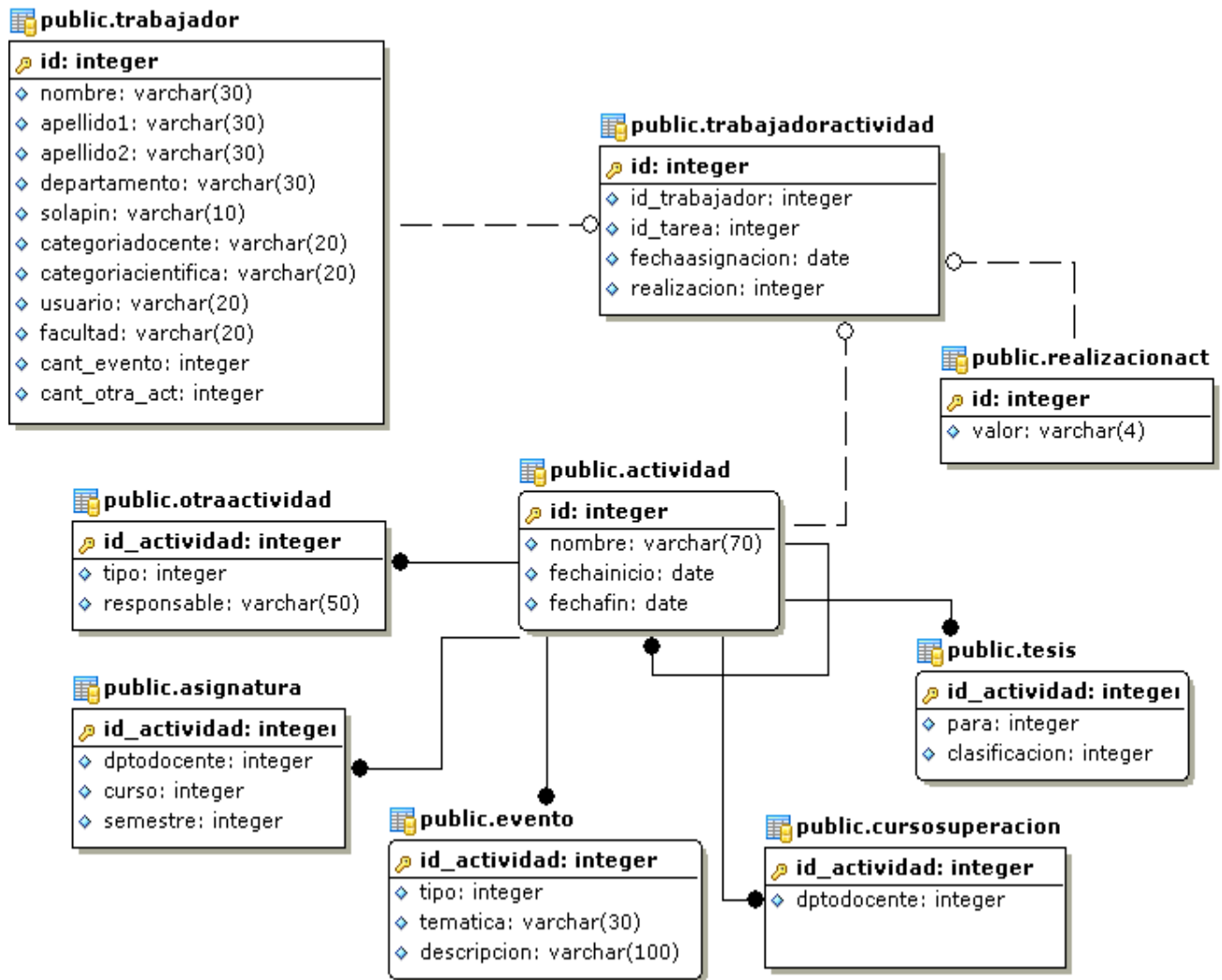


Figura 5: Diseño de la Base de Datos.

### Conclusiones Parciales

En este capítulo se realizó el análisis del sistema, se definieron los estilos y patrones usados con el objetivo de lograr una mayor organización en los elementos que conforman la aplicación. Se modeló el diseño de la Base de Datos, así como la descripción de cada tabla clave para el desarrollo de la Base de Datos.





### CAPÍTULO 4: IMPLEMENTACIÓN Y PRUEBA

#### Introducción

En el presente capítulo se documentan las fases de implementación y prueba según lo propone la metodología de desarrollo XP. Se desglosan las HU en tareas con el objetivo de facilitar el trabajo de los desarrolladores. Se muestran los resultados de las pruebas de aceptación realizadas a la aplicación.

#### 4.1. Implementación

En esta fase XP propone tener en cuenta una serie de aspectos como son la disponibilidad del cliente y el desarrollo en pareja para lograr mayores resultados en la implementación del software.

Disponibilidad del cliente: El cliente formó parte del equipo de desarrollo, describió las HU, guió la toma de decisiones, aprobó las versiones del producto y verificó el cumplimiento de los objetivos trazados.

Desarrollo en pareja: Toda la implementación fue realizada por dos personas que trabajaron en forma conjunta.

##### 4.1.1. Algoritmo de Inteligencia Artificial. Primero el mejor.

En cada paso que realiza el algoritmo primero el mejor selecciona el nodo más prometedor que se haya generado hasta ese momento. Esto se puede conseguir definiendo una función heurística que sea apropiada para resolver el problema en cuestión. La función heurística realiza una estimación de los méritos de cada uno de los nodos que se van generando esto permite que el algoritmo examine primero los caminos más prometedores.

**Función heurística:**  $F(n) = h(n)$ , donde  $n$  es el nodo que se esté analizando.

$H(n) =$  Coste estimado del camino más corto desde un nodo hasta el objetivo.

Cuando el directivo a cargo desee asignar una actividad de último momento y no sepa cuál de sus trabajadores es el mejor para realizarla el sistema será capaz de proponerle aquel que pudiera ser el más idóneo teniendo en cuenta para ello la cantidad de tareas de este tipo que el trabajador haya resuelto satisfactoriamente, mientras mayor sea el valor que tome la función mayor calidad tendrá la respuesta del sistema, para ello se definió como función heurística:

$F(\text{trabajador}) = h(\text{trabajador})$  donde  $h$  representa la cantidad de actividades que ha resuelto el trabajador del tipo de la que se desee asignar.

Para la selección de los trabajadores que se van a sugerir se realiza una preselección entre todos los trabajadores del centro, y se escogen aquellos que hayan realizado al cien por ciento actividades similares o iguales a la que se desee asignar. El sistema propondrá aquel que más actividades haya realizado satisfactoriamente, siendo este el que más experiencia pudiera tener para resolver la tarea. Los directivos del centro no solo podrán trabajar de manera más fácil y organizada sino que también ahorrarán tiempo a la hora de asignar las actividades.

### 4.1.2. Tareas de ingeniería

Para llevar a cabo la correcta implementación de las HU descritas por el cliente se deben definir por parte del equipo de desarrollo las TI (Tareas de Ingeniería) que se realizarán en cada una de las iteraciones.

Las TI también conocidas como tareas de implementación permiten a los desarrolladores obtener un nivel de detalle más avanzado que el que propicia las HU.

El desarrollo del software se planificó en tres iteraciones de trabajo. Para la implementación de las HU correspondientes a la primera iteración se definieron las siguientes TI:

#### Tareas de ingeniería para la iteración #1:

##### HU#1

Tarea de Ingeniería	
Número de Tarea:1	Número de la HU:1
Nombre de Tarea: Autenticación	
Tipo de Tarea: Desarrollo	
Fecha Inicio:20/2/2012	Fecha Fin:22/2/2012
Programador Responsable: Alexei Sánchez Figueroa	
Descripción: El sistema muestra la opción para que el usuario pueda autenticarse en el sistema, ya sea por el dominio UCI o con una cuenta de usuario local.	

Tabla 27: Tarea de ingeniería: Autenticación

HU#2

Tarea de Ingeniería	
Número de Tarea:2	Número de la HU:2
Nombre de Tarea: Crear Cuenta de Usuario Local	
Tipo de Tarea: Desarrollo	
Fecha Inicio:22/2/2012	Fecha Fin:24/2/2012
Programador Responsable: Alexei Sánchez Figueroa	
Descripción: El sistema muestra la opción para que el administrador registre un nuevo usuario local	

Tabla 28: Tarea de ingeniería: Crear Cuenta de Usuario Local

Tarea de Ingeniería	
Número de Tarea:3	Número de la HU:2
Nombre de Tarea: Modificar Cuenta de Usuario Local	
Tipo de Tarea: Desarrollo	
Fecha Inicio:24/2/2012	Fecha Fin:26/2/2012
Programador Responsable: Alexei Sánchez Figueroa	
Descripción: El sistema muestra la opción para que el administrador pueda modificar los datos de la cuenta.	

Tabla 29: Tarea de ingeniería: Modificar Cuenta de Usuario Local

Tarea de Ingeniería	
Número de Tarea:4	Número de la HU:2
Nombre de Tarea: Eliminar Cuenta de Usuario Local	
Tipo de Tarea: Desarrollo	
Fecha Inicio:24/2/2012	Fecha Fin:26/2/2012
Programador Responsable: Alexei Sánchez Figueroa	
Descripción: El sistema muestra la opción para que el administrador pueda eliminar un usuario local.	

Tabla 30: Tarea de ingeniería: Eliminar Cuenta de Usuario Local

HU#3

Tarea de Ingeniería
---------------------

<b>Número de Tarea:</b> 5	<b>Número de la HU:</b> 3
<b>Nombre de Tarea:</b> Registrar Trabajador	
<b>Tipo de Tarea:</b> Desarrollo	
<b>Fecha Inicio:</b> 26/2/2012	<b>Fecha Fin:</b> 26/2/2012
<b>Programador Responsable:</b> Maité Benítez Ortiz	
<b>Descripción:</b> El sistema muestra la opción para que el usuario se autentique en el sistema, una vez registrado el usuario en el sistema se le muestra en la interfaz principal una opción en donde el usuario registrará los datos de su cuenta.	

Tabla 31: Tarea de ingeniería: Registrar trabajador

Tarea de Ingeniería	
<b>Número de Tarea:</b> 6	<b>Número de la HU:</b> 2
<b>Nombre de Tarea:</b> Modificar Trabajador	
<b>Tipo de Tarea:</b> Desarrollo	
<b>Fecha Inicio:</b> 26/2/2012	<b>Fecha Fin:</b> 27/2/2012
<b>Programador Responsable:</b> Maité Benítez Ortiz	
<b>Descripción:</b> El sistema muestra la opción para que el trabajador pueda modificar los datos de su cuenta.	

Tabla 32: Tarea de ingeniería: Modificar trabajador

Tarea de Ingeniería	
<b>Número de Tarea:</b> 7	<b>Número de la HU:</b> 2
<b>Nombre de Tarea:</b> Eliminar Trabajador	
<b>Tipo de Tarea:</b> Desarrollo	
<b>Fecha Inicio:</b> 27/2/2012	<b>Fecha Fin:</b> 27/2/2012
<b>Programador Responsable:</b> Maité Benítez Ortiz	
<b>Descripción:</b> El sistema muestra la opción para que el administrador pueda eliminar un usuario del sistema.	

Tabla 33: Tarea de ingeniería: Eliminar Trabajador

**HU#5**

Tarea de Ingeniería	
<b>Número de Tarea:</b> 11	<b>Número de la HU:</b> 5
<b>Nombre de Tarea:</b> Asignar Actividad	
<b>Tipo de Tarea:</b> Desarrollo	
<b>Fecha Inicio:</b> 2/3/2012	<b>Fecha Fin:</b> 4/3/2012
<b>Programador Responsable:</b> Maité Benítez Ortiz	
<b>Descripción:</b> El sistema muestra la opción para que el controlador pueda asignar una actividad al trabajador.	

Tabla 34: Tarea de ingeniería: Asignar Actividad

**HU # 8**

Tarea de Ingeniería	
<b>Número de Tarea:</b> 12	<b>Número de la HU:</b> 8
<b>Nombre de Tarea:</b> Registrar un Evento	
<b>Tipo de Tarea:</b> Desarrollo	
<b>Fecha Inicio:</b> 4/3/2012	<b>Fecha Fin:</b> 5/3/2012
<b>Programador Responsable :</b> Alexei Sánchez Figueroa	
<b>Descripción:</b> El sistema muestra la opción para que el controlador pueda registrar un evento.	

Tabla 35: Tarea de ingeniería: Registrar evento

Tarea de Ingeniería	
<b>Número de Tarea:</b> 13	<b>Número de la HU:</b> 8
<b>Nombre de Tarea:</b> Modificar Evento	
<b>Tipo de Tarea:</b> Desarrollo	
<b>Fecha Inicio:</b> 5/3/2012	<b>Fecha Fin:</b> 6/3/2012
<b>Programador Responsable:</b> Alexei Sánchez Figueroa	
<b>Descripción:</b> El sistema muestra la opción para que el controlador pueda modificar los datos de un evento.	

Tabla 36: Tarea de ingeniería: Modificar Asignatura

Tarea de Ingeniería	
Número de Tarea:14	Número de la HU:8
Nombre de Tarea: Eliminar Evento	
Tipo de Tarea: Desarrollo	
Fecha Inicio:6/3/2012	Fecha Fin:7/3/2012
Programador Responsable: Alexei Sánchez Figueroa	
Descripción: El sistema muestra la opción para que el controlador pueda eliminar un evento	

Tabla 37: Tarea de ingeniería: Eliminar Evento

### Tareas de ingeniería para la iteración #2

Las HU definidas para la iteración 2 fueron divididas en las siguientes TI. (Ver Anexo 1)

### Tareas de ingeniería para la iteración #3

Las HU definidas para la iteración 3 fueron divididas en las siguientes TI. (Ver Anexo 2)

#### 4.1.3. Diagrama de Despliegue

El modelo de despliegue define la arquitectura física del sistema por medio de nodos interconectados. Estos nodos son elementos hardware sobre los cuales pueden ejecutarse los elementos software.[38]

Los nodos y conexiones del modelo de despliegue y la asignación de los objetos activos a los nodos pueden mostrarse en diagramas de despliegue.[38] Los diagramas de despliegue muestran cómo se configuran las instancias de los componentes de software para su ejecución en los nodos y la comunicación entre ellos.

La configuración propuesta estará conformada por un servidor de aplicación y de base de datos para almacenar toda la información del sistema. Este a su vez se conectará con el servidor LDAP para verificar la autenticidad del usuario. Las máquinas clientes a su vez estarán conectadas con el servidor de aplicación, las cuales podrán tener incorporadas una impresora para obtener una copia dura de los Planes de Trabajo.

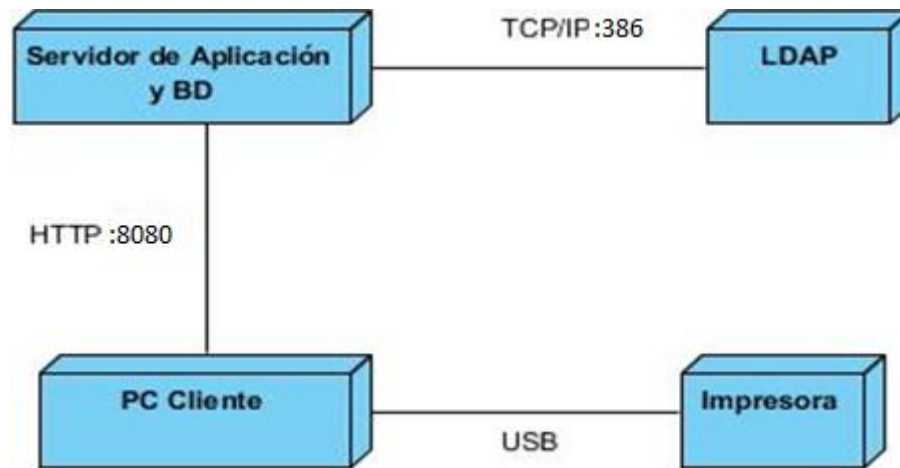


Figura 6: Diagrama de Despliegue.

### 4.2. Validación de la Propuesta

Una vez concluida la implementación del sistema se desarrollaron una serie de pruebas para comprobar que se cumplieron los requerimientos descritos por el cliente. Uno de los pilares fundamentales de la metodología XP es la fase de Prueba. XP divide esta fase en dos grupos: pruebas unitarias, encargadas de verificar el código y diseñada por los programadores, y pruebas de aceptación o pruebas de funcionalidades destinadas a evaluar si al final de una iteración se consiguió la funcionalidad requerida diseñadas por el cliente.[39]

Generalmente las pruebas unitarias las realizan los mismos desarrolladores para comprobar que las funcionalidades implementadas sean las esperadas en función de los requisitos del sistema.

Las pruebas de aceptación resultan de gran importancia, ya que significan la conformidad del cliente con el producto desarrollado, representan el fin de una iteración y el principio de la otra.

#### 4.2.1. Pruebas Unitarias

En las metodologías tradicionales, la fase de pruebas, incluyendo la definición de las pruebas, usualmente son realizadas sobre el final del proyecto, o sobre el final del desarrollo de cada módulo. La metodología XP propone un modelo inverso, en el que, lo primero que se escribe son las pruebas que el sistema debe pasar. Luego, el desarrollo debe ser el mínimo necesario para pasar las pruebas previamente definidas.

Las pruebas a las que se refiere esta práctica, son las pruebas unitarias, realizadas por los desarrolladores. La definición de estos test al comienzo, condicionan o “dirigen” el desarrollo.[40]

Las pruebas unitarias son una de las piedras angulares de XP. Todos los módulos deben de pasar las pruebas unitarias antes de ser liberados o publicados.[40]

Durante el proceso de desarrollo del sistema se realizaron pruebas unitarias para ir comprobando el correcto funcionamiento y el avance del sistema. Las pruebas fueron realizadas por los desarrolladores a medida que se implementaba la solución. Al finalizar cada iteración las funcionalidades fueron probadas por el desarrollador que no había participado en su implementación, de esta manera ambos programadores comprobaban el correcto funcionamiento de los requisitos implementados.

En el transcurso de las pruebas realizadas los errores más notables pertenecían a las HU “Enviar Notificación” y “Proponer trabajador para una actividad”, aunque también se encontraron varios errores en las validaciones de los datos, en la gestión de los nomencladores, y en la organización de los datos en las interfaces de usuario. Los errores eran corregidos en el mínimo de tiempo para lograr que se entregara el producto con la menor cantidad de no conformidades posibles.

### **4.2.2. Pruebas de Aceptación**

Las pruebas de aceptación son creadas a partir de las HU. Una HU puede tener tantas pruebas de aceptación como sean necesarias para comprobar su correcto funcionamiento, la HU no se considera completa hasta que no supera todas las pruebas de aceptación.

Estas pruebas deben realizarse con la mayor rapidez posible, para que los desarrolladores puedan realizar los cambios necesarios y el tiempo de desarrollo no se vea afectado gravemente.

Para las pruebas de aceptación se utilizaron planillas en formato Excel, donde serán probadas cada una de las funcionalidades que debe cumplir el sistema, los casos de pruebas se encuentran estructurados en tres partes:

- **Presentación:** Se define quien elaboró y quien aprobó el caso de prueba así como la versión de la prueba entre otros datos.
- **Casos de Prueba:** Se describe el caso de prueba. Se describen las condiciones de ejecución de la funcionalidad que se está probando, así como los resultados del sistema para cada posible escenario.



- Variables: Se describen cada una de los campos con los que cuenta la funcionalidad.

A continuación se muestran los resultados de las pruebas de aceptación realizadas a las HU pertenecientes a la primera iteración del software.

### 4.2.3. Resultados de las pruebas de aceptación para cada iteración de prueba

A las HU pertenecientes a la iteración 1 se les desarrollaron 3 iteraciones de pruebas y como resultado se detectaron 5 no conformidades para la primera iteración, 3 para la segunda y para la 3 iteración fueron mitigadas todas las no conformidades.

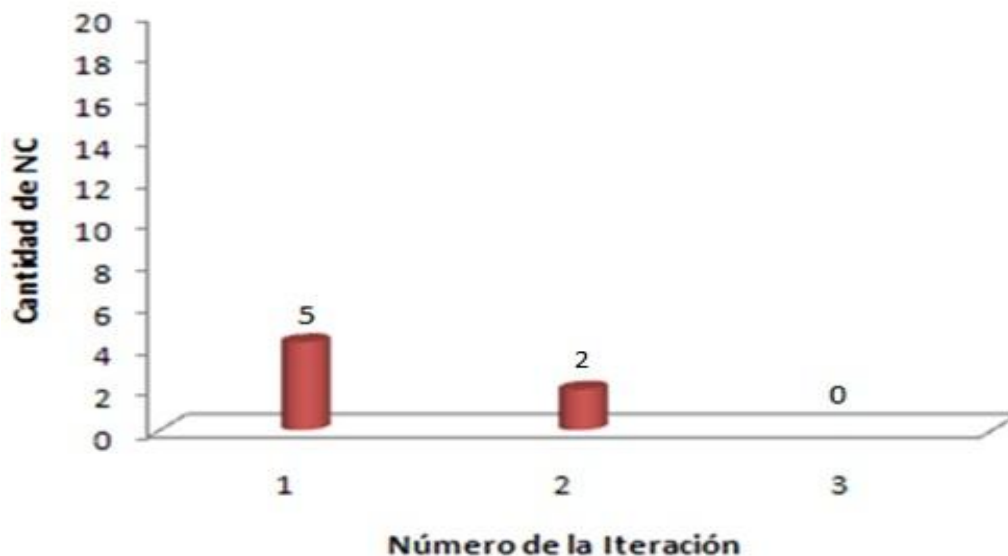


Figura 7. Resultados de las pruebas de aceptación.

Resultados de las pruebas de aceptación para la iteración 2 y 3. (Ver anexo 3)

### Conclusiones Parciales

En este capítulo se generaron los artefactos propuestos por la metodología XP para las fases de Implementación y Prueba. Se realizaron las tareas de ingeniería que dieron solución a todas las HU logrando una mayor organización y rapidez en el desarrollo del sistema. Se muestran los resultados de las pruebas unitarias y de aceptación realizadas para comprobar que se cumplieron todos los objetivos trazados.

### CONCLUSIONES

Una vez concluida la investigación y la implementación del sistema que informatiza la gestión de los Planes de Trabajo del Centro de Telemática se obtuvieron los siguientes resultados:

- Se realizó un análisis de los sistemas similares existentes de los cuales se obtienen ejemplos de cómo podrían solucionarse algunas funcionalidades que se requieren para la aplicación.
- Se formalizó un estudio de las principales metodologías, lenguajes y herramientas que se consideraron factibles para el desarrollo del sistema.
- Se realizó un estudio de técnicas de Inteligencia Artificial que permitieran al jefe del centro asignar las actividades con mayor rapidez y facilidad.
- Se realizó todo el proceso de desarrollo del software siguiendo las fases de la metodología XP, lo cual queda plasmado en el presente documento.

Como resultado se logró el desarrollo de un software en el que se da cumplimiento a las especificidades de los objetivos propuestos. Con el valor fundamental de simplificar la demora que produce el procesamiento manual de la información, evitar posibles errores en la elaboración de los Planes de Trabajo, mejorar la gestión de las actividades que se realizan por parte de los trabajadores del centro así como elevar la calidad en el cumplimiento de las mismas.

### RECOMENDACIONES

Los objetivos generales de este trabajo fueron alcanzados, pero durante su desarrollo surgieron varias ideas que serían recomendables tener en cuenta para su futuro perfeccionamiento:

- Aplicar el software en otros centros de desarrollo para facilitar el trabajo de los directivos con la gestión de los Planes de Trabajo.
- Agregarle al sistema nuevas funcionalidades, que permitan asignarles las evaluaciones trimestrales a los trabajadores.
- Extender la investigación hacia otras áreas, o sea que los Planes de Trabajo se le puedan gestionar no solo a los trabajadores docentes sino a todo el personal que labora en una facultad, ejemplo: secretarias, personal de limpieza.

### REFERENCIAS BIBLIOGRÁFICAS

1. *Resolución No. 60/11*, C. General, Editor. La Habana, 1 de marzo de 2011.
2. Lic. Maibel Garma de la Osa, L.A.R.E., Dr. Julio Vidal Larramendi., “*Planificador: Manejador de planes de trabajo*”.
3. López, M., *Susbsistema de control DE RR.HH.* 2011.
4. *Definición.* 2011 [cited 2011 diciembre 2]; Available from: <http://definicion.de/plan-de-trabajo/>.
5. Mejía., L.R.C., *Planeación Estratégica de Recursos Humanos.* 2004.
6. *¿Qué son los sistemas de gestión?* 2011 [cited 2011 diciembre 2]; Available from: <http://www.bsigroup.com.mx/es-mx/Auditoria-y-Certificacion/Sistemas-de-Gestion/De-un-vistazo/Que-son-los-sistemas-de-gestion/>.
7. RÍOS, J.G.S. (2008) *Gestión humana : tendencias y perspectivas.* **24 No. 107.**
8. S.L, B.I. 2011 [cited 2011 diciembre 4]; Available from: <http://recursosenweb.com/basecamp-herramienta-de-trabajo-colaborativo-para-proyectos-empresariales/>.
9. Kobylanski, W. 2008 [cited 2011 diciembre 4]; Available from: <http://www.htmlife.com/archivos/basecamp/>.
10. Lauzán, O.C. (2010) *Cómo sacar el mejor provecho de los planes de trabajo.*
11. Dirección Técnica, U. *Portal del Centro de Telemática (TLM).* 2010-2011 [cited 2011 diciembre 7]; Available from: <http://portal.dt.prod.uci.cu/projects/politico/wiki/GESPRO>.
12. Elaine Rich, K.K., *Inteligencia artificial.* Segunda Edición ed. 1994: Impreso y Revistas,SA(IMPRESA). 693.
13. Pérez, D.R.B. *Curso de Métodos de Solución de Problemas para la Inteligencia Artificial.*
14. *Entorno Virtual de Aprendizaje.* 2010 [cited 2011 diciembre 6]; Available from: <http://eva.uci.cu/mod/resource/view.php?id=3376&subdir=/Principal>.
15. Rich, E., *Inteligencia Artificial. Segunda Edición.* 1994, Madrid.
16. Murcia, U.d. *Ingeniería del software. Metodologías de desarrollo.* 2010 [cited 2011 Diciembre 7]; Available from: <http://www.um.es/docencia/barzana/IAGP/IAGP2-Metodologias-de-desarrollo.html>.
17. Acuña, K.B., *Selección de metodologías de desarrollo para aplicaciones web en la facultad de informática de la Universidad de Cienfuegos.* . 2009, Universidad de Cienfuegos.
18. Patricio Letelier, M.C.P., *Métodologías ágiles para el desarrollo de software: eXtreme Programming (XP).*

19. José H. Canós, P.L.y.C.P., *Métodologías Ágiles en el Desarrollo de Software*: Valencia.
20. Vallecillo, L.F.y.A., *Una Introducción a los Perfiles UML*. 2010: Málaga.
21. Alfaro, F.M., *Herramientas Case*.
22. Technologies, E., *ER/Studio Modelado de Datos Empresarial*. 2009.
23. López, E.S., *Frameworks para desarrollo de aplicaciones Web*.
24. Niño, E., *Frameworks de desarrollo sobre páginas JSP*  
Autor.: 2012.
25. Clara Cala Rivero Directora , Á.L.R., José Angel Barroso *Reflexiones sobre el Framework de desarrollo del Consejo Superior de Investigaciones Científicas*, 2011.
26. Ing. Maikel Pereira Ojeda , I.H.R.G. (2010) *Desnormalización de la Base de Datos Relacional en favor del rendimiento, para aplicaciones de gestión de información con Hibernate*.
27. Jesús M. González Barahona, J.S.y.G.R., *Introducción al software libre*, UOC, Editor. 2007, 2007.
28. *¿Qué es un Sistema Gestor de Bases de Datos o SGBD*. 2011 [cited 2011 diciembre 4]; Available from: <http://www.cavsi.com/preguntasrespuestas/que-es-un-sistema-gestor-de-bases-de-datos-o-sgbd/>.
29. Yeni Morgado Sánchez, A.C.A., *Guía para la personalización de PostgreSQL 8.4*. 2010, Universidad de las Ciencias Informáticas: Cuba.
30. Carlos A Zuluaga Toro, C.G.S.T., Elkin A Rodríguez (2009) *Laboratorio de Automática Vía Internet*. **2, Num 2**.
31. Kew, N., *Desarrollo de módulos y aplicaciones con Apache*, A. MULTIMEDIA, Editor. 2008: España.
32. Fuentes, M.P.S. (2010) *Propuesta de Framework para servicios basados en localización en Cuba*.
33. Rosa Ma Durante Lerate, P.R.Q., Leandro Pastrana González, Noelia Sales Montes, *Sistemas para el Control de Versiones*, U.D. CÁDIZ, Editor. 2008, 2008.
34. *Centro de Telemática Facultad 2. Descripción de Estructuras v2*.
35. Orjuela Duarte, A.R.C., Mauricio, *Las Metodologías de Desarrollo Ágil como una Oportunidad para la Ingeniería del Software Educativo*, U.N.d. Colombia, Editor. 2008: Colombia.
36. Erich Gamma, R.H., Ralph Johnson, John Vlissides, *Patrones de Diseño. Elementos de software orientado a objetos reutilizables*. 2003, España.
37. Reynoso, C.B., *Introducción a la Arquitectura de Software*. Marzo de 2004: BUENOS AIRES.

38. Ivar Jacobson, G.B., James Rumbaugh., *El proceso unificado de desarrollo de software*. 2000, España.
39. Maite Rodríguez Corbea, M.O.P., *La metodología XP aplicable al desarrollo del software en Cuba*. 2007, Universidad de las Ciencias Informáticas.
40. Joskowicz, I.J., *Reglas y Prácticas en eXtreme Programming*. 2008.

**BIBLIOGRAFÍA**

1. Planificador. [cited 2011 diciembre 2]; Available from: <http://ceces.upr.edu.cu/index.php/component/content/article/38-tecnologia-educativa/111-planificador.html>.
2. Centro de Telemática Facultad 2. Descripción de Estructuras v2.
3. Entorno Virtual de Aprendizaje. 2010 [cited 2011 diciembre 6]; Available from: <http://eva.uci.cu/mod/resource/view.php?id=3376&subdir=/Principal>.
4. Definición. 2011 [cited 2011 diciembre 2]; Available from: <http://definicion.de/plan-de-trabajo/>.
5. ¿Qué son los sistemas de gestión? 2011 [cited 2011 diciembre 2]; Available from: <http://www.bsigroup.com.mx/es-mx/Auditoria-y-Certificacion/Sistemas-de-Gestion/De-un-vistazo/Que-son-los-sistemas-de-gestion/>.
6. ¿Qué es un Sistema Gestor de Bases de Datos o SGBD. 2011 [cited 2011 diciembre 4]; Available from: <http://www.cavsi.com/preguntasrespuestas/que-es-un-sistema-gestor-de-bases-de-datos-o-sgbd/>.
7. Resolución No. 60/11, C. General, Editor. La Habana, 1 de marzo de 2011.
8. Acuña, K.B., Selección de metodologías de desarrollo para aplicaciones web en la facultad de informática de la Universidad de Cienfuegos. . 2009, Universidad de Cienfuegos.
9. Alfaro, F.M., Herramientas Case.
10. Carlos A Zuluaga Toro, C.G.S.T., Elkin A rodríguez (2009) Laboratorio de Automática Vía Internet. **2,Num 2**.
11. Clara Cala Rivero Directora , Á.L.R., José Angel Barroso Reflexiones sobre el Framework de desarrollo del Consejo Superior de Investigaciones Científicas, 2011.
12. Dirección Técnica, U. Portal del Centro de Telemática (TLM). 2010-2011 [cited 2011 diciembre 7]; Available from: <http://portal.dt.prod.uci.cu/projects/politico/wiki/GESPRO>.
13. Elaine Rich, K.K., Inteligencia artificial. Segunda Edición ed. 1994: Impreso y Revistas,SA(IMPRESA). 693.
14. Erich Gamma, R.H., Ralph Johnson,John Vlissides, Patrones de Diseño.Elementos de software orientado a objetos reutilizables. 2003, España.
15. Fuentes, M.P.S. (2010) Propuesta de Framework para servicios basados en localización en Cuba.

16. Ing. Maikel Pereira Ojeda , I.H.R.G. (2010) Desnormalización de la Base de Datos Relacional en favor del rendimiento, para aplicaciones de gestión de información con Hibernate.
17. Ivar Jacobson, G.B., James Rumbaugh., El proceso unificado de desarrollo de software. 2000, España.
18. Jesús M. González Barahona, J.S.y.G.R., Introducción al software libre, UOC, Editor. 2007, 2007.
19. José H. Canós, P.L.y.C.P., Metodologías Ágiles en el Desarrollo de Software: Valencia.
20. Kew, N., Desarrollo de módulos y aplicaciones con Apache, A. MULTIMEDIA, Editor. 2008: España.
21. Kobylanski, W. 2008 [cited 2011 diciembre 4]; Available from: <http://www.htmlife.com/archivos/basecamp/>.
22. Lauzán, O.C. (2010) Cómo sacar el mejor provecho de los planes de trabajo.
23. López, E.S., Frameworks para desarrollo de aplicaciones Web.
24. López, M., Subsistema de control DE RR.HH. 2011.
25. Maite Rodríguez Corbea, M.O.P., La metodología XP aplicable al desarrollo del software en Cuba. 2007, Universidad de las Ciencias Informáticas.
26. Murcia, U.d. Ingeniería del software. Metodologías de desarrollo. 2010 [cited 2011 Diciembre 7]; Available from: <http://www.um.es/docencia/barzana/IAGP/IAGP2-Metodologias-de-desarrollo.html>.
27. Niño, E., Frameworks de desarrollo sobre páginas JSP, Autor:. 2012.
28. Oktaba, H., Introducción a Patrones. 2010: Facultad de Ciencias, UNAM.
29. Orjuela Duarte, A.R.C., Mauricio, Las Metodologías de Desarrollo Ágil como una Oportunidad para la Ingeniería del Software Educativo, U.N.d. Colombia, Editor. 2008: Colombia.
30. Patricio Letelier, M.C.P., Metodologías ágiles para el desarrollo de software: eXtreme Programming (XP).
31. Pérez, D.R.B. Curso de Métodos de Solución de Problemas para la Inteligencia Artificial.
32. Rich, E., Inteligencia Artificial. Segunda Edición. 1994, Madrid.
33. RÍOS, J.G.S. (2008) Gestión humana : tendencias y perspectivas. **24 No. 107**.
34. Rosa Ma Durante Lerate, P.R.Q., Leandro Pastrana González, Noelia Sales Montes, Sistemas para el Control de Versiones, U.D. CÁDIZ, Editor. 2008, 2008.
35. S.L, B.I. 2011 [cited 2011 diciembre 4]; Available from: <http://recursosenweb.com/basecamp-herramienta-de-trabajo-colaborativo-para-proyectos-empresariales/>.
36. Technologies, E., ER/Studio Modelado de Datos Empresarial. 2009.

37. Vallecillo, L.F.y.A., Una Introducción a los Perfiles UML, C.d.T.E.-M. (SPAIN), Editor: Málaga.
38. Vallecillo, L.F.y.A., Una Introducción a los Perfiles UML. 2010: Málaga.
39. Yeni Morgado Sánchez, A.C.A., Guía para la personalización de PostgreSQL 8.4. 2010, Universidad de las Ciencias Informáticas: Cuba.
40. Hernán., S.M., Diseño de una Metodología Ágil de Desarrollo de Software. 2004, Universidad de Buenos Aires.
41. Víctor Theoktisto, Patrones de Diseño. 2000.
42. Solís., M.C., Una explicación de la programación extrema (XP). 2003.
43. KING, C.B.G., Hibernate in Action. 2005.
44. Patrick Peak, N.H., Hibernate Quickly. 2006.
45. Mann, K.D., Java server Faces in Action. 2005.