

**Universidad de las Ciencias Informáticas**

**Facultad 7**



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

**Desarrollo del Módulo de Registro de Investigaciones del  
Sistema Integral de Gestión Administrativa de la Facultad 7**

**Autoras:** Danay Ramírez Barroso  
Yanelis Castrillo González

**Tutores:** Ing. Darling Darías Pérez  
Ing. Alejandro Arias Naranjo

**Ciudad de La Habana, Julio del 2010**

**“Año 52 de la Revolución”**



*“La enseñanza que deja huellas no es la que se hace  
de cabeza a cabeza, sino de corazón a corazón”*

*Howard G. Hendricks*



### DATOS DE CONTACTO

**TUTOR: Ing. Darling Darías Pérez.** Graduada de Ingeniera en Ciencias Informáticas en la UCI en el curso 2008. Profesora Adiestrada. Ha impartido las asignaturas de Gestión de Software, Sistemas de Bases de Datos y Preparación para las Pruebas de Nivel de Ingeniería de Software. Perteneció a la Subdirección de Gestión de la Calidad como administradora de la misma. Actualmente es subdirectora de la Subdirección de Investigaciones y Postgrado del CESIM.

**Correo electrónico:** [ddarias@uci.cu](mailto:ddarias@uci.cu) **Teléfono:** (07) 837- 3711

**TUTOR: Ing. Alejandro Arias Naranjo.** Graduado de Ingeniero en Ciencias Informáticas en la UCI en el curso 2007. Profesor Instructor. Ha impartido las asignaturas de Algoritmización, Introducción a la Programación y Programación 1. Perteneció al grupo GPI (Grupo de Procesamiento de Imágenes) donde se ejerció como jefe de proyecto. Pertenece actualmente al vicedecanato de investigaciones de la Facultad 7, desempeñado las funciones de asesor de investigación.

**Correo electrónico:** [aariasn@uci.cu](mailto:aariasn@uci.cu) **Teléfono:** (07) 837- 3160



---

## RESUMEN

El presente trabajo surge ante la necesidad de viabilizar el proceso de gestión de la información relacionada con la gestión investigativa en la Facultad 7 de la Universidad de las Ciencias Informáticas. Esta investigación propone la implementación de un sistema que permita viabilizar la gestión de la información generada en los procesos relacionados con la gestión investigativa en la Facultad 7 de la Universidad de las Ciencias Informáticas.

La solución propuesta formará parte del Sistema Integral de Gestión de Administrativa de la facultad, por lo que se definieron las tecnologías y herramientas a emplear para la implementación de los módulos. Como metodología de desarrollo se define Rational Unified Process. Como lenguaje de modelado UML. Como herramienta de modelado Enterprise Architect. Como gestor de bases de datos se tiene PostgreSQL. Se utiliza Python como lenguaje de programación, Django como marco de desarrollo web y WingIDE como entorno de desarrollo. Además se implementa el patrón de arquitectura Model View Template.

Una vez desarrollado el sistema facilitará el trabajo de la Dirección de Investigaciones y los profesores de la facultad, además de generar reportes que son necesarios para entregar de manera rápida y sencilla la información en la Dirección de Investigaciones de la Universidad de las Ciencias Informáticas. También se espera llevar un control más exacto de las investigaciones realizadas por cada profesor, y a su vez de los avances que en este ámbito tiene la facultad.

## PALABRAS CLAVES

Investigación, Evento, Publicaciones, Premios.



**TABLA DE CONTENIDOS**

INTRODUCCIÓN.....	1
CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA .....	6
1.1. Sistemas automatizados existentes vinculados al campo de acción .....	6
1.2. Sistemas automatizados existentes en la Universidad de las Ciencias Informáticas .....	10
1.3. Metodología de desarrollo de software, lenguaje y herramienta para el modelado .....	12
1.3.1. RUP .....	13
1.3.2. UML .....	14
1.3.3. Enterprise Architect (EA).....	16
1.4. Sistema Gestor de Base de Datos .....	17
1.4.1. PostgreSQL.....	17
1.5. Lenguaje de programación y entorno de desarrollo .....	18
1.5.1. Python .....	19
1.5.2. Django .....	20
1.5.3. Wing IDE .....	20
CAPÍTULO 2: CARACTERÍSTICAS DEL SISTEMA .....	22
2.1. Objeto de Estudio .....	22
2.1.1. Flujo actual de los procesos involucrados en el campo de acción .....	22
2.1.2. Análisis crítico de la ejecución de los procesos .....	23
2.2. Información que se maneja .....	24
2.3. Objeto de automatización .....	24
2.4. Modelo del Negocio .....	25
2.4.1. Actores del Negocio .....	25
2.4.2. Trabajadores del Negocio .....	25
2.4.3. Diagrama de Casos de Uso del Negocio .....	26
2.4.4. Descripción Textual de los Casos de Uso del Negocio .....	26
2.4.5. Diagrama de Actividades .....	27
2.4.6. Diagrama de Clases del Modelo de Objetos .....	28
2.5. Especificación de los Requerimientos del Software .....	29
2.5.1. Requerimientos funcionales.....	29
2.5.2. Requerimientos no funcionales.....	30
2.6. Modelo de Casos de Uso .....	32




---

2.6.1.	Actores del Sistema .....	32
2.6.2.	Diagrama de Casos de Uso del Sistema.....	33
2.6.3.	Especificación de Casos de Usos del Sistema .....	33
2.6.4.	Casos de uso expandidos.....	38
CAPÍTULO 3: DISEÑO DEL SISTEMA.....		42
3.1.	Descripción de la arquitectura. Fundamentación.....	42
3.2.	Principios de diseño.....	44
3.2.1.	Diagrama de clases del diseño.....	44
3.2.2.	Diagramas de Interacción en el diseño .....	46
3.2.3.	Descripción de las clases del diseño.....	48
CAPÍTULO 4: IMPLEMENTACIÓN Y PRUEBA.....		50
3.1.	Modelo de Datos .....	50
3.1.1.	Descripción de las tablas .....	51
3.2.	Implementación.....	53
3.2.1.	Diagrama de despliegue .....	54
3.2.2.	Diagrama de Componentes .....	55
3.3.	Seguridad.....	56
CONCLUSIONES .....		58
RECOMENDACIONES.....		59
REFERENCIAS.....		60
BIBLIOGRAFÍA.....		61
ANEXOS .....		<b>¡Error! Marcador no definido.</b>



---

## INTRODUCCIÓN

La investigación es un proceso de trascendental importancia, nace en la vida cotidiana del ser humano y está presente en toda su existencia, pues el hombre tiene una tendencia natural a buscar el sentido de las cosas. Es por ello que existen diversos tipos de investigación, desde las más elementales y cotidianas, hasta la investigación científica con características propias de eficacia superior.

La investigación científica, es una indagación mucho más rigurosa. Muchos autores coinciden con la idea de que es un estudio sistemático y objetivo de un tema claramente delimitado y debe contar con una constante disciplina y responsabilidad. El éxito de la investigación depende en gran medida de no suprimir o descuidar las etapas que la componen.

Para el sistema de educación superior es vital el proceso de investigación científica, ya que se encarga de exigir a las universidades la producción de conocimiento científico socialmente válido capaz de generar soluciones creativas en las múltiples áreas del quehacer social. Por lo que, la investigación científica en la Educación Superior tiene una doble función: contribuye en la formación del profesional y es además una vía para resolver problemas que se presentan en la sociedad. (1)

Es aproximadamente tres décadas atrás cuando las universidades empiezan a crear estructuras dedicadas a la gestión de la investigación, con distintas funciones y dependencias orgánicas. A su vez en América Latina las instituciones de educación superior comienzan a enfrentarse a grandes retos, como la insuficiencia de recursos en el sector educacional y el empobrecimiento de la actividad científica. Esto provoca la fuga de cerebros hacia centros de producción desarrollados y privados e induce un retraso en la aparición del proceso de gestión de la investigación en estos países.

A nivel mundial existen múltiples modelos de gestión de la investigación, cada uno orientado a necesidades particulares de cada institución. Para un proceso de gestión de la investigación balanceado y eficaz se debe proponer un modelo que se adecúe a los requerimientos del centro y que además le permita al investigador realizar su tarea, con mayor facilidad.

En sentido general todos los modelos están enfocados a cumplir un grupo de funciones, como son: asumir toda la gestión administrativo-económica de las investigaciones, gestionar el estado de



---

superación del personal, gestionar el estado de los proyectos investigativos, normativas legales y al mismo tiempo debe ser el medio de comunicación para aclarar y atender dudas o trámites.

El modelo cubano de la nueva universidad científica y tecnológica tiene como estrategia clave la flexibilidad organizativa, la cooperación nacional e internacional y la búsqueda de recursos materiales y financieros que permitan su desarrollo. El Ministerio de Educación Superior (MES) hoy en día está centrado en consolidar la interrelación entre la formación de profesionales y la vida económica, política y social del país, a partir del concepto de integración de la docencia con la producción y la investigación en su concepción más amplia y más rica. (2)

A raíz de esta situación es que surge la Universidad de las Ciencias Informáticas (UCI), como un nuevo modelo de universidad. Esta se ha convertido en el principal pilar de la industria cubana del software, junto a otras empresas como Desoft y Softel; haciendo grandes aportes al país tanto en el ámbito informático como en el económico.

En la actualidad, la UCI cuenta con una Dirección de Investigaciones que trabaja por el buen desempeño de la actividad científica del centro. Sus principales esfuerzos están dirigidos a organizar las investigaciones que se realizan en este centro e incorporarlo al sistema científico nacional, lo que permitirá estimular la participación de profesores y estudiantes en esta actividad. Además cuenta con un Consejo Científico, el cual se apoya en las facultades para la planificación de eventos o actividades de carácter investigativo y guiar en los procesos de investigación de su facultad.

La Facultad 7 como parte de la estructura docente-productiva-investigativa, mantiene dos líneas de investigación: procesamiento de imágenes y señales e informatización del sector de la salud, donde se desarrollan productos nacionales e internacionales. La misma está organizada en departamentos productivos y docentes. Estos últimos atienden los recursos humanos y el currículum vitae de cada uno de los profesores que lo integran, los cuales son la materia prima para asegurar cada uno de los procesos.

Los departamentos son atendidos por el vicedecano de investigación, el asesor de investigaciones y el subdirector de investigaciones y postgrado del Centro Especializado de Soluciones Informáticas Médicas (CESIM). Los mismos rinden cuentas a la Dirección de Investigaciones a nivel de Universidad. La información de cada profesor es muy diversa, incluye su categoría docente y científica; las



---

actividades en las que participa, ya sea de superación, docencia o producción; participación en eventos y misiones dentro y fuera del país; publicaciones en revistas nacionales e internacionales, etc.

Actualmente existe una herramienta orientada a la gestión de varios de estos procesos investigativos, pero la misma está enfocada a nivel de universidad y funcionalmente no cumple con los requisitos para adecuarla a la estructura interna de la facultad; trayendo consigo que la información que se genera en la facultad se mantenga actualizada en hojas de cálculo o en documentos separados según lo decida cada departamento o vicedecanato.

Solo una persona interactúa con la información, demorando todo el proceso de actualización de los datos. Además, no existe visibilidad o accesibilidad entre una información y otra, de modo que se facilite la toma de decisiones sobre un profesor en particular; esto hace que haya que recurrir a informaciones viejas, muchas veces desactualizadas, para generar un nuevo informe.

Lo mencionado anteriormente, trae como consecuencia que no se pueda tener un control estricto de todas las actividades realizadas por un profesor en el campo investigativo. Por lo que no se tiene centralizado el estado de superación, la participación en eventos, talleres u otras actividades. Esta información además de las publicaciones, artículos y experiencias bibliográficas del docente no se socializa; haciendo engorroso el trabajo de los jefes de departamento y del vicedecano de investigaciones. Todo esto trae como resultado final un desconocimiento de las capacidades del equipo de trabajo, asignación errónea de tareas y falta de alineación entre personas, estrategia de superación y adquisición de conocimiento.

Por lo antes planteado se identifica como **problema a resolver** ¿Cómo viabilizar el proceso de gestión de la información relacionada con la gestión investigativa en la Facultad 7 de la Universidad de las Ciencias Informáticas?

Este problema se enmarca en el **objeto de estudio**: el proceso de gestión administrativo en la Universidad de las Ciencias Informáticas. El objeto delimita el **campo de acción**: el proceso de gestión de la información que se genera como resultado de las investigaciones realizadas en la Facultad 7 de la Universidad de las Ciencias Informáticas.



---

Para la solución del problema se plantea como **objetivo general**: Desarrollar un sistema informático que permita viabilizar la gestión de la información que se genera en los procesos relacionados con la gestión investigativa en la Facultad 7 de la Universidad de las Ciencias Informáticas

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

- Analizar los procesos del negocio asociados a la gestión de la información relacionada con la gestión investigativa en la facultad.
- Realizar un análisis acerca de los sistemas informáticos existentes a nivel nacional e internacional referentes a la gestión de la investigación.
- Asimilar la arquitectura definida para el desarrollo del Sistema Integral de Gestión Administrativa.
- Modelar los procesos actuales de la gestión de la información relacionada con la gestión investigativa en la facultad.
- Analizar las necesidades de funcionamiento de la aplicación describiendo la especificación de Requerimientos de Software.
- Realizar el modelado de casos de uso del sistema.
- Diseñar el sistema informático utilizando la arquitectura definida para el desarrollo del Sistema Integral de Gestión Administrativa.
- Realizar el diseño de la base de datos.
- Escribir un manual de usuario que facilite el trabajo y la navegación por la herramienta.

Con el desarrollo de del Módulo de Registro de Investigaciones del Sistema Integral de Gestión Administrativa de la Facultad 7, se obtendrán una serie de **resultados esperados**:

- Facilitar el trabajo de los directivos de la Dirección de Investigaciones y de los profesores, los cuales podrán obtener el estado de publicaciones, eventos, premios, categorías y otros datos personales de cada personal de la facultad.
- Establecer y crear por departamento (docente y/o productivo) un expediente con informaciones referentes a investigaciones, cursos, eventos, publicaciones, conferencias y talleres que permita el control y seguimiento del avance de la investigación de cada profesor.
- Generar reportes sobre temas de interés como: cantidad de publicaciones (UCI, Nacional, Internacional), cantidad de eventos participados (UCI, Nacional, Internacional), cantidad de premios obtenidos por la facultad, entre otros reportes.



---

Y se obtendrá además como **beneficios**:

- Centralizar la información referente a temas investigativos lográndose la viabilidad y accesibilidad de los datos.
- Garantizar la calidad y la rapidez en la entrega y recepción de información.
- Obtener estadísticas en tiempo real a partir de la información generada por el sistema.

El trabajo está estructurado en cuatro capítulos que se describen a continuación:

**CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA:** En este capítulo se realiza un análisis de los sistemas existentes a nivel internacional de gestión de la investigación, haciendo una valoración acerca del por qué no se utilizó ninguna de las herramientas encontradas. Además se describen las tecnologías a tener en cuenta para modelar e implementar el sistema.

**CAPÍTULO 2: CARACTERÍSTICAS DEL SISTEMA:** Se describe el proceso de negocio que tiene lugar en la Dirección de Investigaciones de la facultad. Se concretan las características del sistema y el objeto de automatización. Se definen y se describen además los actores y trabajadores del negocio, los casos de uso del negocio y la descripción de los mismos. Los Requerimientos funcionales y no funcionales del sistema también están explícitos, así como la definición de los actores y casos de uso del sistema, con la especificación de los más significativos.

**CAPÍTULO 3: ANÁLISIS Y DISEÑO DEL SISTEMA:** Se hace referencia a los modelos de análisis y diseño, que incluyen: los diagramas de clases del análisis y del diseño; los diagramas de interacción de los casos de uso del sistema más críticos; la descripción de las clases del diseño, así como patrones de diseño utilizados.

**CAPÍTULO 4: IMPLEMENTACIÓN Y PRUEBA:** Aborda la implementación, donde se mencionarán aquellos aspectos relacionados con la construcción de la solución propuesta, se modelan los diagramas de componentes y despliegue; se describen los estándares de diseño, codificación y el tratamiento de errores para darle solución al sistema propuesto.



En este capítulo se realizará un estudio detallado de las herramientas más utilizadas a nivel mundial para la gestión investigativa que pudieran ser empleadas en el desarrollo de la aplicación web. Se abordará todo lo relativo al funcionamiento del departamento de investigaciones de la universidad.

Debido a que la solución propuesta formará parte de un sistema integral constituido por varios módulos, la dirección de la facultad definió un conjunto de tecnologías y herramientas, las cuales están especificadas en el documento de arquitectura de la facultad. En este capítulo se analizan las principales características de las mismas.

### **1.1. Sistemas automatizados existentes vinculados al campo de acción**

La transformación de las universidades en centros con alto grado de complejidad que comienzan a involucrarse activamente en el desarrollo de investigaciones y programas de postgrado, aparejado a las posibilidades que ofrecen actualmente las nuevas tecnologías de la información y las comunicaciones; va a establecer una tendencia a la creación de aplicaciones web para apoyar la gestión investigativa que se realiza en las universidades. Estas aplicaciones se caracterizan por almacenar documentos y por ser esencialmente informativas, por lo que se hace imprescindible la aparición de los Sistemas de Gestión Investigativa (SGI).

Los SGI no son más que sistemas, que permiten la actualización de la información que se genera en las universidades, ya sea acerca de sus proyectos de investigación; los resultados de los investigadores en términos de publicaciones y eventos científicos fundamentalmente; oportunidades de financiación, entre otros. A continuación se mostrará un estudio realizado a sistemas utilizados por varias universidades que pudieran ser de ayuda para la implementación de la solución propuesta.



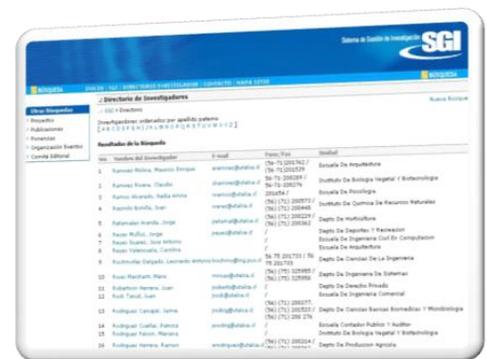
La **Universidad de Talca**, cuenta con una aplicación web que se encarga de apoyar a la Dirección de Investigación. La misma brinda información acerca de los objetivos que persiguen; los diversos fondos que poseen (se refiere a cada uno de los bienes que brinda la Dirección para contribuir en el apoyo, ya sea para académicos jóvenes post-graduados, académicos con experiencia comprobada en investigación, desarrollo de actividades académicas de carácter regional, nacional o internacional, entre otros). Posee un archivo donde se puede acceder a las Memorias<sup>1</sup> de diversos años y además cuenta con un estudio estadístico acerca del número de publicaciones de la universidad, por año.



**Figura 1:** Sistema de la Dirección de Investigaciones – Universidad de Talca, Chile.

Debido al carácter únicamente informativo de esta aplicación, se planteó la conveniencia de crear un sistema de información que satisfaga requerimientos tanto de nivel operativo, táctico como estratégico de las autoridades institucionales, sus investigadores y de los diferentes actores interesados en el trabajo investigativo y sus resultados, en particular el empresarial y gubernativo. Para que el sistema cumpliera con las expectativas planteadas se consideró esencial implementarlo sobre una plataforma Web, planteándose la necesidad de desarrollar un sitio Web que soportara el Sistema de Gestión de la Investigación (SGI). (3)

El entorno del **SGI** de la Universidad de Talca gira alrededor de una página de búsqueda donde se puede hacer búsquedas por proyectos, publicaciones, ponencias, eventos y además cuenta con un directorio de investigadores, que brinda información variada acerca de los mismos como: proyectos a los que está vinculado, publicaciones que posee y otros elementos de interés como eventos o ponencias en los que ha participado. Del mismo modo posee un espacio de inscripción para optar por los fondos con los que cuenta la Universidad.



**Figura 2:** Sistema de Gestión de las Investigaciones – Universidad de Talca.

<sup>1</sup> Resume las actividades investigativas desarrolladas por una Institución durante un período de tiempo específico.

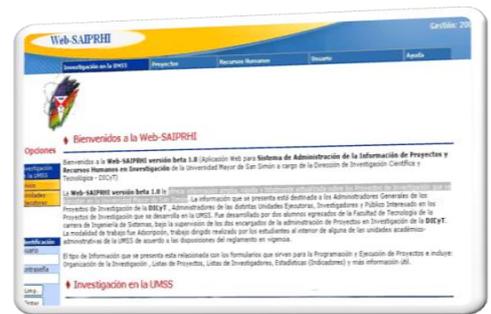


**La Universidad Mayor de San Simón en Bolivia (UMSS)** se propuso en el año 2003 llevar a cabo un Programa de Cooperación de la Investigación Científica, siendo uno de sus resultados el diseño de un sistema para fortalecer la gestión de las actividades de investigación en dicha universidad. Este es un sistema fundamentalmente informacional que se encarga de publicar documentos referidos a temas investigativos, información acerca de convenios internacionales y posee además un conjunto de vínculos a todas las herramientas y sistemas asociados a la Dirección de Investigaciones Científica y Tecnológica (DICyT).



**Figura 3:** Sistema de la Dirección de Investigación Científica y Tecnológica (DICyT) - Universidad de Talca.

**La Web-Saiprhi versión beta 1.0** (Aplicación Web para Sistema de Administración de la Información de Proyectos y Recursos Humanos en Investigación a cargo del DICyT) ofrece información amplia, rápida y totalmente actualizada sobre los Proyectos de Investigación y sobre el personal académico que participa en la planificación y ejecución de los proyectos de la UMSS. Por lo que este sistema sería igualmente de gran ayuda para el desarrollo de la solución propuesta.



**Figura 4:** Sistema de Administración de la Información de Proyectos y recursos Humanos en Investigación - UMSS.

La Web-Saiprhi de la UMSS al igual que el SGI de la Universidad de Talca fue construida enteramente para suplir las necesidades de estas universidades y en función de sus niveles de organización. No son adaptables a otras instituciones; por lo que solo se utilizarán como ejemplo para la implementación de algunas funcionalidades del sistema propuesto.

**UNIVERSITAS XXI – INVESTIGACIÓN** forma parte de una Solución Integral para la gestión universitaria puesta en marcha por la Oficina de Cooperación Universitaria (OCU) como la realización de un proyecto de las universidades públicas de Alcalá, Carlos III de Madrid, Castilla-La Mancha, Salamanca y Valladolid, a la que posteriormente se unieron el Grupo Santander (2003) y la Universidad Rey Juan Carlos de Madrid (2004). Esta solución se ha difundido en una comunidad que agrupa en la actualidad a 976 universidades de once países: España, Argentina, Brasil, Chile,



Colombia, México, Perú, Portugal, Puerto Rico, Uruguay y Venezuela gracias a la OCU, siendo este Sistema de Gestión de la Investigación uno de los más usados en la comunidad latinoamericana. (4)

UNIVERSITAS XXI - INVESTIGACIÓN, es una solución integral para la Gestión de la Investigación. Un sistema construido íntegramente en tecnología Web. Se ha diseñado pensando en el investigador y por ello trata de favorecer su labor principal, descargándole de las tareas administrativas que lleva asociadas, cada actividad investigadora.



Figura 5: Sistema Universitas XXI - Módulo Investigación.

El investigador tiene un portal propio, donde encontrará de forma personalizada, todos los datos del módulo que de alguna manera se relacionen con él, como son: sus publicaciones, sus proyectos, sus colaboraciones, sus empresas, etc. Además, es el componente donde se asienta el currículum. El sistema registra los datos de cada investigador y de sus actividades y el currículum es tan sólo una salida electrónica o impresa. Por eso, es independiente de cualquier formato existente: se adapta a todos y de forma automática emite el deseado en cada momento por el investigador sin ningún trabajo adicional.

UNIVERSITAS XXI – INVESTIGACIÓN responde a la concepción de un producto escalable en diferentes instancias, que resuelve las necesidades de los investigadores, las de las universidades las de los organismos públicos de investigación y las de las Comunidades.

Tiene como principales características:

- Alto nivel de flexibilidad para contemplar los distintos procedimientos y particularidades que se dan en las universidades.
- Ofrece información descentralizada y en su justa medida a los distintos sectores interesados: investigadores, empresas, gestores de la investigación, ciudadanos.
- Proporciona una ayuda incuestionable al personal investigador con funcionalidades tales como la gestión curricular, gestión de convocatorias, seguimiento de su actividad investigadora, etc.
- Incorpora elementos que facilitan la búsqueda de financiación, lo que es fundamental en un entorno cada día más competitivo.



- Permite realizar una gestión integral de los proyectos de investigación contemplando las particularidades concretas de cada tipo de actividad investigadora. (5)

Un valor añadido de UNIVERSITAS XXI – INVESTIGACIÓN radica en su integración con el resto de los sistemas universitarios, dentro de UNIVERSITAS XXI:

- Con UNIVERSITAS XXI - ECONÓMICO en la gestión económica de los proyectos, tanto a nivel de gastos como de ingresos.
- Con UNIVERSITAS XXI - RECURSOS HUMANOS en los flujos de información que producen, en ambos sentidos, el personal investigador y los becarios.
- Con UNIVERSITAS XXI - ACADÉMICO en lo referente asignaturas, estudiantes, etc.

Entre las principales desventajas se puede encontrar que: eventualmente se deben instalar los restantes módulos para garantizar una completa gestión universitaria. Utiliza Oracle como gestor de base de datos. Las licencias Universitas XXI se tramitan a través de donación del Grupo OCU, las licencias de Oracle son embebidas en los productos UNIVERSITAS XXI y están dentro de los costos de implantación. Los cuales promedian entre los \$60.000.000 y \$120.000.000.

Es necesario añadir que el módulo UNIVERSITAS XXI – INVESTIGACIÓN no posee una arquitectura adaptable al Sistema Integral de Gestión Administrativa del cual formará parte el Módulo de Registro de Investigaciones. Por todas estas razones solo se utilizará este sistema como modelo para el desarrollo del software.

## **1.2. Sistemas automatizados existentes en la Universidad de las Ciencias Informáticas**

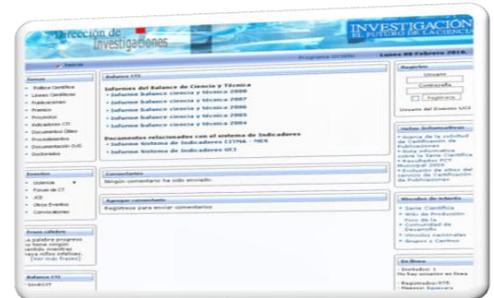
La creación de la Universidad de las Ciencias Informáticas, con una fuerte base tecnológica y un amplio perfil productivo, donde el estudiante se forma desde la producción; ha garantizado una amplia capacidad investigativa que facilita las condiciones necesarias para que la universidad se convierta en el centro de la enseñanza superior de mayor fortaleza científica del país en el área de la informática y tenga un peso importante dentro del sistema científico cubano. Todo esto impone a la universidad una participación destacada en las investigaciones dirigidas a buscar soluciones a los problemas que se presentan en el proceso productivo, lo que se corresponde con el concepto de universalización de la enseñanza y el principio de la unidad de la docencia, la investigación y la producción.



Ante esta situación y debido a la necesidad de una organización acertada de las investigaciones universitarias surge la Dirección de Investigaciones la cual trabaja incansablemente por la organización y el buen desempeño de la actividad científica del centro. Sus principales esfuerzos están dirigidos a organizar las investigaciones en la Universidad estimulando la participación de profesores y estudiantes, a establecer una alianza estratégica con el MES, el CITMA y el Polo Científico e incorporar el centro al sistema científico nacional.

Para lograr los objetivos establecidos, la Dirección de Investigaciones se apoya en una aplicación web que en sentido general facilita la divulgación de información. Entre sus principales funcionalidades se destacan:

Brindar toda la información referente a la Política Científica de la Universidad, Líneas de Investigación, Doctorados y Procedimientos. Publicar documentación útil para realizar una publicación o documentación de proyectos y temas afines, normas de publicación para la serie interna de la UCI y notas informativas. Publicar todos los premios obtenidos en el ámbito universitario o eventos a nivel nacional e internacional. Publicar informes de balance de ciencia y técnica y las convocatorias de los diversos eventos que se realizan en la universidad, así como sus resultados.



**Figura 6:** Sistema de la Dirección de Investigaciones de la UCI.

Para realizar el Balance de Ciencia, Tecnología e Innovación, la Dirección de Investigaciones se apoya en un Sistema de Indicadores (SindiCIT) el cual brinda la posibilidad de llevar un control de la información de manera cuantificada por facultades, con los cuales se arrojan resultados estadísticos que dan una idea de cómo se encuentra la institución.



**Figura 7:** Sistema de Indicadores de Ciencia y Tecnología e Innovación.

Actualmente se encuentran en desarrollo tres trabajos de diploma, en apoyo a la Dirección de Investigación. Uno de ellos abarcará la implementación de un sistema que gestionará todos los eventos científicos realizados en la Universidad. Otro trabajará en la implementación de un sistema que controle todos los datos referentes a las publicaciones científicas, el mismo no estará diseñado en función de la gestión de publicaciones, sino solo gestionar metadatos relacionadas con estas.



El tercer trabajo de diploma se encargará del perfeccionamiento del Sistema de Indicadores de Ciencia Tecnología e Innovación y su integración con los módulos de **Gestión de Premios Curriculares y Publicaciones Científicas** y **Gestión de Eventos Científicos**.

Estas investigaciones trabajarán enfocadas a solucionar problemas directamente relacionados con la Dirección de Investigaciones de la Universidad, por lo que no van a resolver la situación problemática planteada en el presente estudio; es necesario añadir que el mismo se basa fundamentalmente en integrar y vincular a los recursos humanos con los elementos mencionados anteriormente, de forma tal que se pueda obtener de un profesor sus publicaciones, los eventos en los que ha participado o los premios que ha alcanzado, entre otras funcionalidades.

Realizado un análisis de los sistemas automatizados relacionados con el campo de acción y los sistemas utilizados por la Universidad, se arribaron a las conclusiones de que ningún sistema resolvería los problemas existentes, por lo que se decide implementar un módulo de Registro de Investigaciones, el cual se encargará de la obtención, registro y transferencia de la información, referente a los temas investigativos en la Facultad 7.

Este módulo formará parte del Sistema Integral de Gestión de Administrativa de la Facultad 7, el cual incluirá una serie de módulos, que viabilizarán todos los procesos administrativos de la facultad, entre los que se encuentran:

- Módulo de Gestión de Recursos Humanos.
- Módulo de Planificación de Actividades.
- Módulo de Gestión de Reuniones.
- Módulo de Gestión de Tesis.
- Módulo de Gestión de Recursos Materiales.

### **1.3. Metodología de desarrollo de software, lenguaje y herramienta para el modelado**

Las metodologías de desarrollo de software son un conjunto de procedimientos, técnicas y ayuda a la documentación para el desarrollo de productos software, en la que se van indicando paso a paso todas las actividades a realizar para lograr informatizar el proceso deseado; indicando qué personas deben participar en el desarrollo de las actividades y el papel que desempeñan. Además, detallan la



información que se debe producir como resultado de una actividad y la información necesaria para comenzarla. (6)

En el mundo existen diversas y variadas metodologías de desarrollo de software, pero no todas se integran a las necesidades requeridas para el buen desarrollo de un software.

## 1.3.1. RUP

Acrónimo de Rational Unified Process (Proceso Unificado de Rational) es un proceso de ingeniería de software, una forma disciplinada de asignar tareas y responsabilidades (quién hace qué, cuándo y cómo) en una organización o equipo de desarrollo de software. Es también un producto desarrollado y mantenido por Rational, actualizado constantemente para tener en cuenta las mejores prácticas de acuerdo con la experiencia. Junto con el Lenguaje Unificado de Modelado UML, constituye la metodología estándar más utilizada para el análisis, implementación y documentación de sistemas orientados a objetos.

Provee un enfoque disciplinado en la asignación de tareas y responsabilidades dentro de una organización de desarrollo. Su meta es asegurar la producción de software de muy alta calidad que satisfaga las necesidades de los usuarios finales, dentro de un calendario y presupuesto predecible.

RUP pretende implementar las mejores prácticas actuales en Ingeniería de Software:

- Desarrollo iterativo del software.
- Administración de requerimientos.
- Uso de arquitecturas basadas en componentes.
- Modelación visual del software.
- Verificación de la calidad del software.
- Control de cambios.

Se caracteriza por ser iterativo e incremental, estar centrado en la arquitectura y guiado por los casos de uso. En lo que se refiere al ciclo de vida de RUP, es una implementación del desarrollo en espiral. Este divide el proceso de desarrollo del software en ciclos, donde estos son la clave que tiene el modelo para crear un proyecto de buena calidad. El ciclo de vida organiza las tareas en fases e iteraciones.



Está compuesto por cuatro fases: Inicio, Elaboración, Construcción y Transición, cada una de ellas compuesta de una o varias iteraciones. A su vez está compuesto por nueve Flujos de Trabajo, seis de ellos de Ingeniería y tres de apoyo: Modelado del negocio, Requerimientos, Análisis y diseño, Implementación, Pruebas y Despliegue, Administración de cambio y configuración, Administración de proyecto y Entorno. En cada una de sus fases se emplean todos los flujos de trabajo pero con diferente énfasis.

Para desarrollar este trabajo se decidió utilizar la metodología RUP, la cual es una de las metodologías pesadas, pues está basada en normas provenientes de estándares seguidos por el entorno de desarrollo y proceso mucho más controlado, con numerosas políticas. Está pensada para adaptarse a cualquier proyecto, divide el proceso de desarrollo, en ciclos de iteración, teniendo un producto final, al concluir cada ciclo y es una de las metodologías más importante para alcanzar un grado de certificación en el desarrollo del software.

## 1.3.2. UML

Acrónimo de Unified Modeling Language (Lenguaje Unificado de Modelado) es el lenguaje de modelado más conocido y utilizado en la actualidad. Es un lenguaje gráfico para visualizar, especificar, construir y documentar un sistema de software. Las propiedades que han hecho de UML un estándar son: la concurrencia, es un lenguaje distribuido y adecuado a las necesidades de conectividad actual y futura. Reemplaza a decenas de notaciones empleadas con otros lenguajes. Modela estructuras complejas. Las estructuras que soporta tienen sus fundamentos en las tecnologías orientadas a objetos, tales como clases, componentes y nodos. Representa el comportamiento del sistema a través de casos de uso, diagramas de secuencia y de colaboración.

Uno de los objetivos de este modelado visual es que sea independiente del lenguaje de implementación, de tal forma que los diseños realizados se puedan implementar en cualquier lenguaje que soporte las posibilidades de UML (principalmente lenguajes orientados a objetos). Con este método formal de modelado se pueden automatizar determinados procesos y permite generar código a partir de los modelos y a la inversa.

**UML 2.0** es la mayor revisión que se le ha hecho a UML desde la versión 1.0. El modelo conceptual ha sido reestructurado completamente y nuevos diagramas han sido incorporados. Los diagramas tradicionales también han sido mejorados. La nueva versión permitirá a los fabricantes de herramientas CASE, proporcionar a los analistas, arquitectos y desarrolladores, herramientas cada vez más



potentes, que les permitan aprovechar mejor los modelos y como consecuencia generar una mayor cantidad de código, reduciendo significativamente el ciclo de desarrollo de sus aplicaciones. (7)

En UML 2.0 se definen una serie de diagramas adicionales a los establecidos en su versión anterior. Compárese las figuras que se muestran a continuación:

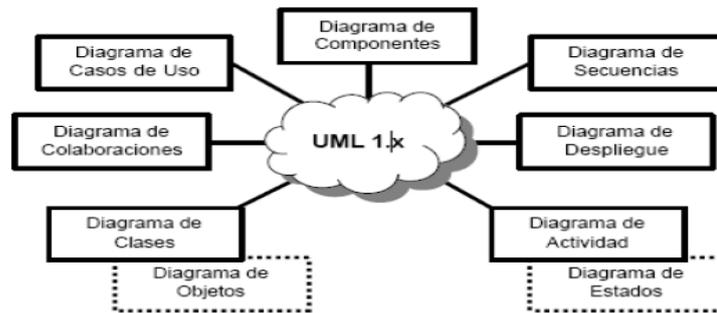


Fig.1 Diagramas de UML 1.0

UML 1.x está constituido por siete diagramas básicos y dos diagramas que constituyen variaciones de dos de los anteriores.

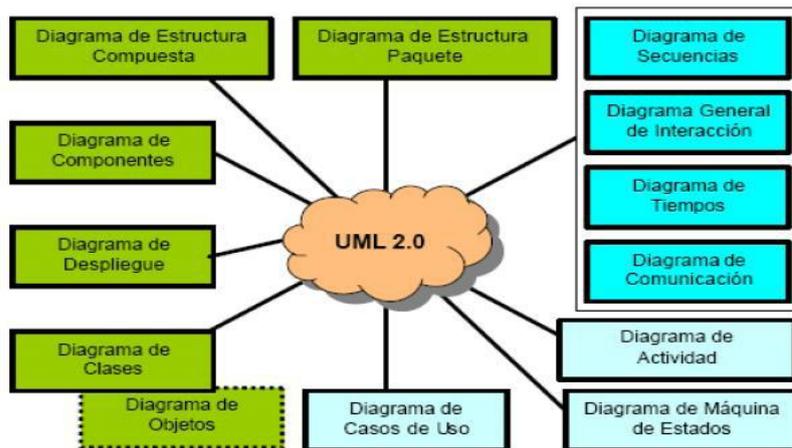


Fig.2 Diagramas de UML 2.0



En el UML 2.0 se definen una serie de diagramas adicionales a los establecidos UML 1.x. El conjunto de diagramas se encuentra organizado en torno a dos categorías: diagramas estructurales (representados en verde) y diagramas dinámicos o de comportamiento (representados en celeste). (8)

La versión que usa este trabajo es la 2.0 debido a las ventajas que ofrece con respecto a su versión anterior 1.0.

### 1.3.3. Enterprise Architect (EA)

Enterprise Architect combina el poder de la última especificación UML 2.0 con alto rendimiento e interfaz intuitiva, para traer modelado avanzado al escritorio y para el equipo completo de desarrollo e implementación.

Características:

- Trazabilidad de extremo a extremo: provee trazabilidad completa desde el análisis de requerimientos hasta los artefactos de análisis y diseño, a través de la implementación y el despliegue. Combinados con la ubicación de recursos y tareas incorporados, los equipos de Administradores de Proyectos y Calidad están equipados con la información que ellos necesitan para ayudarles a entregar proyectos en tiempo.
- Velocidad, estabilidad y buen rendimiento: El Lenguaje Unificado de Modelado provee beneficios significativos para ayudar a construir modelos de sistemas de software rigurosos y donde es posible mantener la trazabilidad de manera consistente. Enterprise Architect soporta este proceso en un ambiente fácil de usar, rápido y flexible.
- Construido sobre las bases de UML 2.0.
- Diseñado para ayudar a construir software robusto y fácil de mantener.
- Permite generar documentación e informes flexibles y de alta calidad.

EA soporta generación e ingeniería inversa de código fuente para muchos lenguajes populares, incluyendo C++, C#, Java, Delphi, VB.Net, Visual Basic y PHP, y también para muchos de los SGBD, incluyendo Oracle, SQLServer, My SQL, PostgreSQL y otros. Modela tablas, llaves, relaciones complejas y perfiles de modelado de datos incluidos, además que permite generar scripts para crear estructuras de base de datos.



Por las potencialidades ofrecidas y además por su alto rendimiento combinado a la última especificación de UML 2.0 se escoge la herramienta Enterprise Architect como herramienta de modelado para llevar la documentación del software.

## 1.4. Sistema Gestor de Base de Datos

El software de base de datos ha experimentado un auge extraordinario a raíz de la progresiva informatización de casi la totalidad de las empresas de hoy día; rapidez, efectividad en los procesos y los grandes flujos de información son las necesidades más apremiantes a la hora de optimizar servicios y productos. Ante esta notable demanda de soluciones informáticas ha surgido un gran cúmulo de gestores de base de datos.

El propósito general de los sistemas de gestión de base de datos es el de manejar de manera clara, sencilla y ordenada un conjunto de datos que posteriormente se convertirán en información relevante, para un buen manejo de datos. Un sistema gestor de base de datos se define como el conjunto de programas que administran y gestionan la información contenida en una base de datos.

Existen distintos objetivos que deben cumplir los SGBD:

Abstracción de la información, independencia del tratamiento de información, redundancia mínima, consistencia, seguridad, integridad, respaldo y recuperación de la información, control de la concurrencia y tiempo de respuesta mínimo.

Las principales funciones que debe realizar un SGBD son:

- La definición de los datos.
- La manipulación de los datos.
- Garantizar la seguridad e integridad de los datos.
- La gestión de las transacciones y el acceso concurrente.

Entre los SGBD más utilizados actualmente se encuentran: SQLServer, MySQL, PostgreSQL y Oracle.

### 1.4.1. PostgreSQL

PostgreSQL es un SGBD orientada a objetos de software libre, publicado bajo la licencia Berkeley Software Distribution (BSD). Posee una serie de características positivas respecto a otros. Por



ejemplo: Posee una gran escalabilidad. Es capaz de ajustarse al número de procesadores y a la cantidad de memoria que posee el sistema de forma óptima, haciéndole capaz de soportar una mayor cantidad de peticiones simultáneas de manera correcta, se dice que ha llegado a soportar el triple de carga de lo que soporta MySQL.

El SGBD Orientado a Objetos PostgreSQL, comenzó su desarrollo sobre los años 80 y actualmente es desarrollado por una comunidad mundial conocida como Grupo Mundial de Desarrollo de PostgreSQL (PGDG). Entre sus principales características se encuentra su destreza para manejar altos niveles de concurrencia mediante un sistema llamado Acceso Concurrente Multiversión (MVCC). Posee una amplia variedad de tipos nativos y permite al usuario crear sus propios tipos de datos. Es multiplataforma y además soporta procedimientos almacenados y *triggers*, los cuales pueden ser escritos usando lenguajes como PL/PgSQL, C++, Java PL/Java web, PL/Python y PL/PHP.

Gracias a su licencia BSD, se permite la utilización del código para ser comercializado. Multiusuario, multiprogramado, con arquitectura cliente-servidor y control de privilegios de acceso.

Las características fundamentales de PostgreSQL son:

- DBMS Objeto-Relacional: Aproxima los datos a un modelo objeto-relacional, y es capaz de manejar complejas rutinas y reglas.
- Altamente extensible: Soporta operadores, funcionales métodos de acceso y tipos de datos definidos por el usuario.
- Integridad Referencial: Soporta integridad referencial, la cual es utilizada para garantizar la validez de los datos de la base de datos.
- Lenguajes Procedurales: Tiene soporte para lenguajes procedurales internos, incluyendo un lenguaje nativo denominado PL/pgSQL. Este lenguaje es comparable al lenguaje procedural de Oracle, PL/SQL. Otra ventaja de PostgreSQL es su habilidad para usar Perl, Python, o TCL como lenguaje procedural embebido.

## 1.5. Lenguaje de programación y entorno de desarrollo

La programación Web, parte de las siglas WWW, que significan World Wide Web o red mundial. Para realizar una página con la programación Web se debe tener claro, uno de los conceptos fundamentales el cual es, el URL (Uniform Resource Locators), sistema con el cual se localiza un recurso dentro de la red, este recurso puede ser una página web, un servicio u otro medio accesible.



## 1.5.1. Python

Python, es un lenguaje ágil, muy potente y fácil de aprender. Cuenta con una gran popularidad a nivel mundial, trayendo consigo que se pueda encontrar una basta documentación sobre el tema. Por ser un lenguaje interpretado, la ejecución del código suele ser más lenta que la puesta en marcha de un archivo binario resultado de la compilación de un lenguaje como puede ser C o C++. Para eliminar esta barrera existe Python.h, una biblioteca que permite la incorporación de código C o C++ como módulos y funciones Python.

Características:

- Lenguaje interpretado o de script: Tiene muchas de las características de los lenguajes compilados, por lo que se podría decir que es semi-interpretado.
- Tipado dinámico: no es necesario declarar el tipo de dato que va a contener una determinada variable, sino que su tipo se determinará en tiempo de ejecución según el tipo de valor que se le asigne, el tipo de esta variable puede cambiar si se le asigna un valor de otro tipo.
- Fuertemente tipado: No se permite tratar a una variable como si fuera de un tipo distinto al que tiene, es necesario convertir de forma explícita dicha variable al nuevo tipo previamente.
- Multiplataforma: El intérprete de Python está disponible en multitud de plataformas (UNIX, Solaris, Linux, DOS, Windows, OS/2, Mac OS, etc.)
- Orientado a objetos: permite la programación imperativa, programación funcional y programación orientada a aspectos.

Aspecto que disminuiría aún más el tiempo de respuesta de los desarrolladores para cada una de las soluciones personalizadas de la distribución GNU/Linux Nova.

Este lenguaje presenta una serie de ventajas que lo hacen muy atractivo, tanto para su uso profesional como para el aprendizaje de la programación, entre las que se destacan:

- Es un lenguaje muy “expresivo”, es decir, los programas son muy compactos, un programa en Python suele ser bastante más corto que su equivalente en lenguajes como C, por muchos Python es considerado un lenguaje de programación de muy alto nivel.
- Es muy legible, la sintaxis de Python es muy elegante y permite la escritura de programas cuya lectura resulta fácil, en comparación con otros lenguajes.
- Puede usarse como lenguaje imperativo procedimental o como lenguaje orientado a objeto.
- Una ventaja fundamental de Python es la gratuidad de su intérprete.



Por todas estas características y ser un lenguaje de fácil aprendizaje, se convierte en el ideal para la implementación de la aplicación que se quiere desarrollar en el marco de la facultad, pues se dispone de muy poco tiempo para la implementación de la misma. Python facilita llevar a cabo este proceso sin perder el foco en el desarrollo. Se utilizará la versión 2.6 pues es la versión más estable y compatible para trabajar con Django.

## 1.5.2. Django

Django es un marco de desarrollo web sobre Python que permite desarrollar rápidamente aplicaciones web. La idea es poder montar sitios nuevos y añadir contenidos de forma muy rápida y dinámica. Posee un diseño perfeccionista, con un código muy legible, una exquisita metodología de desarrollo y una perfecta documentación, además de una amplia comunidad de desarrollo. Muy escalable (habilidad para: manejar el crecimiento continuo de trabajo de manera fluida, y estar preparado para hacerse más grande sin perder calidad en los servicios ofrecidos).

Por la gran facilidad y el ahorro de tiempo que brinda Django cuando se utiliza asociado con el lenguaje de programación Python, es que se decide trabajar con este entorno de desarrollo en su versión 1.1.

## 1.5.3. Wing IDE

Wing IDE es un entorno integrado de desarrollo especialmente diseñado para el lenguaje de programación Python, disponible en Linux y en Windows. Permite el desarrollo rápido de aplicaciones de plataforma cruzada para escritorio, web y empresariales. Wing IDE se enfoca en incrementar la productividad y la calidad del código, especialmente en proyectos complejos con requerimientos cambiantes. Proporciona a los desarrolladores un completo editor de código fuente, depurador, explorador de código y muchas otras herramientas diseñadas específicamente para su uso con Python.

Entre las principales características del Wing IDE se encuentran:

- Auto completado de código.
- Asistente de codificación.
- Identación automática.
- Búsquedas en múltiples archivos.
- Búsquedas en todo el disco.



- Búsquedas con expresiones regulares.
- Soporte de CVS, Subversión.
- Sintaxis coloreada.
- Evaluación de archivos o selecciones de archivo.
- Gratuitos para desarrollos Open Source.

Estas características son las que definen el por qué del uso de Wing IDE como entorno de desarrollo en la implementación del software en su versión 3.2.

El presente capítulo ha posibilitado conocer y adentrarse en el mundo actual de la gestión investigativa en las universidades del mundo. Se identificaron un grupo de resultados de otros sistemas que persiguen el mismo objetivo de informatización, cada uno de ellos con características comunes pero con marcadas diferencias con la universidad debido al complejo nivel organizativo que presenta cada facultad; por lo que solo se utilizarán estos sistemas como ejemplo o guía para el desarrollo de la solución propuesta. Además, se asimilaron las principales tendencias, tecnologías y herramientas definidas por la facultad que se utilizarán tanto en el modelado y la implementación del software, como para la generación de documentos.



# CAPÍTULO 2

## CARACTERÍSTICAS DEL SISTEMA

Es de vital importancia antes de llevar a cabo el proceso de desarrollo de un software conocer todos los aspectos y particularidades del entorno donde se desempeñará. Los posibles procesos que se realizan y que se llevan a cabo en el ambiente deben tener una clara visión de la problemática y buscar soluciones factibles y precisas. Es imprescindible una comunicación efectiva entre los usuarios y el equipo de proyecto para llegar a un entendimiento de lo que hay que hacer, ya que esa es la clave del éxito de un software.

El modelo del negocio, posibilita obtener una visión más clara del proceso, por lo que en este capítulo se abordarán temas relacionados con el negocio, así como todos los elementos referentes a la especificación de los Requerimientos de software y del sistema.

Como paso inicial para el desarrollo del software se realizó un estudio detallado de las actividades que se realizan en la Dirección de Investigaciones de la Facultad 7. Las mismas ayudarán a obtener los requerimientos necesarios, posibilitando identificar los casos concretos que debe tener el sistema, la relación que debe existir entre la ingeniería de software y el negocio, con el fin de aclarar el enfoque que quiere tener el cliente con el software, más un conjunto de artefactos para iniciar la modelación de dicha propuesta de solución al problema presentado.

### 2.1. Objeto de Estudio

#### 2.1.1. Flujo actual de los procesos involucrados en el campo de acción

La Dirección de Investigaciones de la Facultad 7 se encarga de llevar el control del currículum de los profesores de la facultad y del CESIM<sup>2</sup>, esto permite tener un estado de todas las actividades relevantes que realiza un profesor, así como las categorías que el mismo presenta, relación de cursos

---

<sup>2</sup> Centro Especializado en Soluciones Informáticas Médicas



de postgrado impartidos y recibidos, proyectos de innovación o producción ejecutados y otras actividades.

El negocio abarcará solamente los elementos del currículum que tengan relación con: las publicaciones (artículo en revista, ponencia en evento, libros, monografías, textos complementarios, etc.); los eventos (nacionales, internacionales o internos de la universidad); relación de reconocimientos, distinciones o premios (Fórum, CITMA, etc.), las categorías científica y docente; el grado científico; tipo de maestrías, doctorados y diplomados.

El currículum le permitirá al Vicedecano de Investigaciones o a su asesor crear reportes; ya sea por ejemplo: número de profesores que participan en un evento determinado, cantidad de premios obtenidos por la facultad en los eventos, cantidad de publicaciones y otro número de reportes necesarios para realizar el Balance Anual.

El Proceso de Crear Currículum es realizado por el Profesor, el cual se encarga de registrar todos los datos de su currículum en una plantilla oficial llamada Currículum Vitae y enviárselo al Jefe de Departamento, el cual recopila la información necesaria para verificar que los datos estén correctamente avalados. En caso de que esté incompleto el currículum, los Jefes de Departamento añaden información al mismo; luego de revisado y completado se la envía al Vicedecano de Investigación o al Asesor que se encarga de archivar toda la información.

La información recopilada de los currículos de cada profesor permite emitir reportes acerca de las publicaciones científicas realizadas por la facultad, los trabajos presentados en eventos y los premios alcanzados. Estos reportes se utilizan en la elaboración del Balance Anual. En caso de que en el currículum existan actividades que no estén correctamente avaladas los Jefes de Departamento reenvían el currículum al profesor para que él mismo lo actualice o se encargue de gestionar el aval de una determinada actividad.

### **2.1.2. Análisis crítico de la ejecución de los procesos**

El proceso de Crear Currículum que se realiza en la actualidad se ve afectado por un grupo de aspectos que entorpecen la obtención de buenos resultados. En muchos de los casos los Jefes de Departamento envían varios documentos no oficiales a la Dirección de Investigaciones de la facultad con información repetida, ya que no existe una estandarización de la documentación, lo que provoca



que la información al no estar centralizada, se encuentre gran parte duplicada, haciéndola más propensa a errores.

Se emplea mucho tiempo en la verificación de la veracidad de los datos; todos los reportes se realizan manualmente, es decir, se verifica por departamentos los profesores que participan en eventos, las publicaciones que tienen y es la Dirección de Investigaciones la encargada de calcular los reportes con los datos requeridos para realizar el Balance Anual.

Actualmente la Dirección de Investigaciones de la facultad se encarga de enviar mediante el Sistema de Indicadores de la Dirección de Investigaciones de la Universidad los reportes mencionados anteriormente, estos permiten calcular una serie de parámetros preestablecidos, los cuales pueden dar una idea de cómo se encuentra la facultad dentro del análisis del centro.

### **2.2. Información que se maneja**

El Currículo Vitae: es un documento para el control y seguimiento de toda la información profesional de un profesor. Posee nombre y apellidos del profesor, datos personales, datos profesionales como: grado científico, categoría docente y científica, centro de trabajo, cargo que ocupa, labor que desempeña actualmente, idiomas que domina. Además, posee la relación de los cursos recibidos, relación de los cursos impartidos como profesor tanto en pregrado como en postgrado, tesis de grado, maestría, doctorado dirigidas y defendidas, relación de los proyectos de investigación, innovación, producción o servicios ejecutados, relación de las publicaciones, relación de reconocimientos, distinciones o premios, descripción de actividades profesionales, producción y servicios y otras actividades académicas relevantes que desenvuelve.

### **2.3. Objeto de automatización**

El sistema propuesto ha sido concebido con el fin de mejorar el proceso de gestión de la información que se genera como resultado de la gestión de las investigaciones que se realizan en la facultad. La información que se incluye es acerca de las publicaciones que posee el profesor, eventos en los que participa y premios obtenidos. Brindará información referente a normas de publicación y resoluciones.

El sistema automatizará el registro de los datos del currículo que se encuentran dentro del alcance de la investigación, en este caso toda la información de publicaciones, eventos y premios referidos



anteriormente. Los mismos podrán ser insertados por el Profesor y modificados por el Jefe de Departamento, Vicedecano de Investigación, Asesor de Investigaciones y Subdirector de Investigaciones y Postgrado del CESIM.

La aplicación posibilitará la obtención de reportes necesarios para la creación de documentos oficiales como el Balance Anual o para insertar los datos correspondientes al Sistema de Indicadores de la Dirección de Investigaciones de la Universidad.

## 2.4. Modelo del Negocio

### 2.4.1. Actores del Negocio

Un actor del negocio es cualquier individuo, grupo, entidad, organización, máquina o sistema de información externo; con los que el negocio interactúa. Lo que se modela como actor es el rol que se juega cuando se interactúa con el negocio para beneficiarse de sus resultados.

Actores Involucrados	
Rol	Función
Profesor	Es el principal responsable de la elaboración del Currículo Vitae donde introduce todos los datos necesarios referente a temas investigativos.

### 2.4.2. Trabajadores del Negocio

Un trabajador del negocio representa a personas o sistemas dentro del negocio que son los que realizan las actividades que están comprendidas dentro de un caso de uso. Estos trabajadores están dentro de la frontera del negocio, son los que en un futuro se convertirán en usuarios del sistema que se quiere construir.

Trabajadores Involucrados	
Rol	Función
Jefe de Departamento	Es el encargado de modificar, aprobar el currículum y enviarlo a la Dirección de Investigaciones de la facultad.

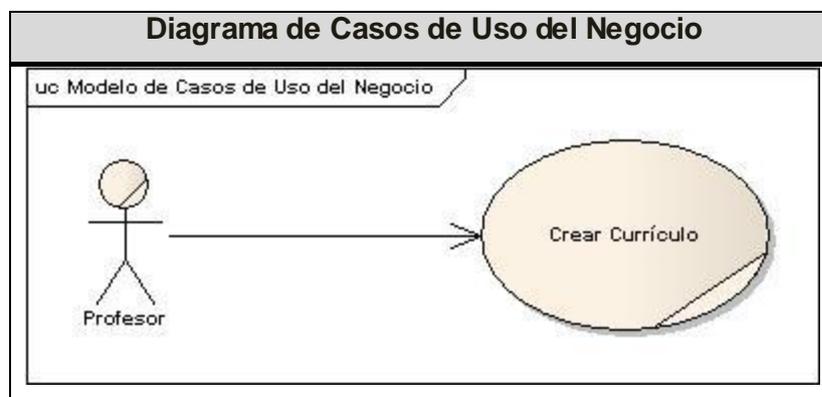


Directivo de Investigaciones	Es el encargado de archivar la información y generar reportes.
------------------------------	--

### 2.4.3. Diagrama de Casos de Uso del Negocio

Un Caso de Uso del Negocio representa un proceso que se ejecuta en el mismo, constituyendo una serie de acciones que muestran un resultado palpable para ciertos actores de dicho negocio.

El diagrama de Casos de Uso del Negocio que se presenta a continuación, representa gráficamente la interacción existente entre el proceso del negocio Crear Currículo y el Actor del Negocio Profesor.



### 2.4.4. Descripción Textual de los Casos de Uso del Negocio

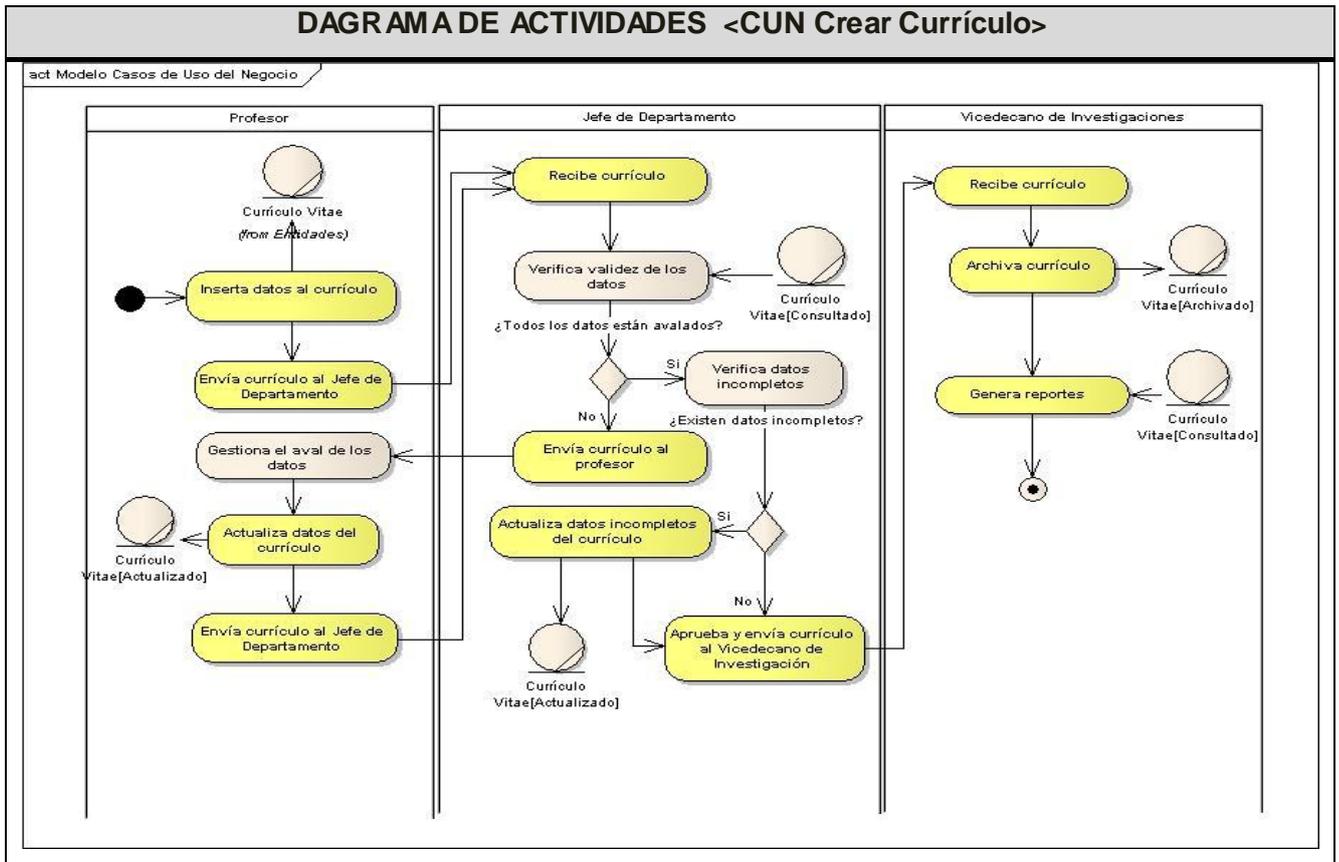
<b>Caso de Uso:</b>	Crear Currículo
<b>Actores:</b>	Profesor (inicia)
<b>Trabajadores:</b>	Jefe de Departamento, Directivo de Investigaciones.
<b>Resumen:</b> El caso de uso se inicia cuando el profesor inserta sus datos al Currículo Vitae, se lo envía al Jefe de Departamento para que lo revise y ese se lo envía al Directivo de Investigaciones para que lo archive.	
<b>Precondiciones:</b>	
<b>Flujo Normal de Eventos</b>	
<b>Actor del Negocio</b>	<b>Respuesta del Negocio</b>



1. Inserta datos de currículum (Currículum Vitae) 2. Entrega currículum	3. El Jefe de Departamento recibe currículum y verifica validez de los datos. 4. Se detectan datos del currículum que no están correctamente avalados. (Ver Flujo Alternativo 1) 5. Se detectan datos relevantes del currículum incompletos. (Ver flujo Alternativo 2) 6. El Jefe de Departamento aprueba el currículum y lo envía al Directivo de Investigaciones. 7. El Directivo de Investigaciones archiva currículum. 8. Genera reportes a partir de los datos de los currículos.
<b>Flujos Alternos</b>	
<b>Flujo Alternativo 1</b>	
<b>Actor del Negocio</b>	<b>Respuesta del Negocio</b>
5. Gestionar aval de datos del currículum. 6. Actualizar datos del currículum.	7. El Jefe de Departamento envía currículum al profesor. 8. Ir al paso 6 del flujo Normal de Eventos.
<b>Flujos Alternos</b>	
<b>Flujo Alternativo 2</b>	
<b>Actor del Negocio</b>	<b>Respuesta del Negocio</b>
	6. Actualiza datos incompletos del currículum. 7. Ir al paso 6 del flujo Normal de Eventos.

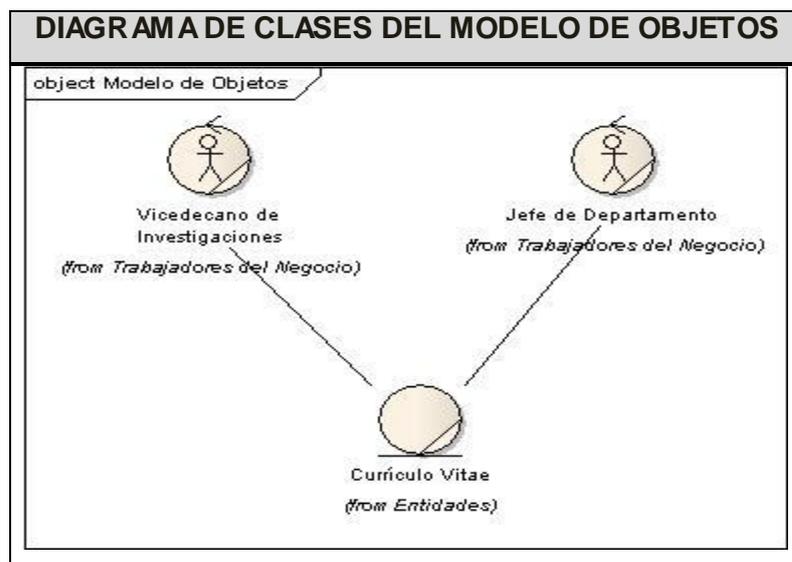
#### 2.4.5. Diagrama de Actividades

Los casos de uso del negocio consisten en secuencias de actividades que, en conjunto, producen algo para el actor del negocio. El proceso consiste en un flujo básico de una o más alternativas de flujos. La estructura del flujo se describe gráficamente con la ayuda de un diagrama de actividad.



**2.4.6. Diagrama de Clases del Modelo de Objetos**

Muestra la participación de los trabajadores, las entidades del negocio y la relación entre ellos. A continuación se muestra el Diagrama de Clases del Modelo de Objetos.





### 2.5. Especificación de los Requerimientos del Software

#### 2.5.1. Requerimientos funcionales

Los Requerimientos Funcionales son capacidades o condiciones con las que debe cumplir el producto a elaborar, no alteran la funcionalidad del software, se mantienen invariables sin importar con que cualidades o propiedades se relacionen.

A partir del estudio y las investigaciones que se realizaron del proceso del negocio, se obtuvieron una serie de requerimientos funcionales:

1. Gestionar Evento
  - 1.1. Buscar evento
  - 1.2. Modificar evento
  - 1.3. Eliminar evento
  - 1.4. Registrar evento
2. Registrar Modalidad de Evento
  - 2.1. Insertar Modalidad
  - 2.2. Eliminar modalidad
3. Registrar Nivel de Evento
  - 3.1. Insertar nivel
  - 3.2. Eliminar nivel
4. Gestionar Trabajo
  - 4.1. Buscar trabajo
  - 4.2. Modificar trabajo
  - 4.3. Verificar trabajo
  - 4.4. Eliminar trabajo
  - 4.5. Registrar trabajo
5. Registrar Tipo de Premio de Trabajo
  - 5.1. Insertar tipo de premio
  - 5.2. Eliminar tipo de premio
6. Gestionar Publicaciones



- 6.1. Buscar publicación
- 6.2. Modificar publicación
- 6.3. Verificar publicación
- 6.4. Eliminar publicación
- 6.5. Registrar publicación
- 7. Registrar Categoría de Publicación
  - 7.1. Insertar categoría
  - 7.2. Eliminar categoría
- 8. Registrar Tipo de Publicación
  - 8.1. Insertar tipo
  - 8.2. Eliminar tipo
- 9. Gestionar Premio
  - 9.1. Registrar premio
  - 9.2. Eliminar premio
- 10. Asignar premio
  - 10.1. Buscar persona
  - 10.2. Asignar premio a persona
  - 10.3. Eliminar asignación de premio
- 11. Registrar Categoría de Premio
  - 11.1. Insertar categoría
  - 11.2. Eliminar categoría
- 12. Subir Certificación
  - 12.1. Ver Certificación
- 13. Generar Reportes
  - 13.1. Generar reporte de trabajos por eventos
  - 13.2. Generar reporte de búsqueda de publicaciones
  - 13.3. Generar reporte de búsqueda de trabajos

### **2.5.2. Requerimientos no funcionales**

Los requerimientos no funcionales son cualidades o propiedades que el producto debe tener. Debe pensarse en las características que hacen al producto o sistema, atractivo, usable, confiable y fácil de usar por el usuario. A continuación se muestran los requerimientos no funcionales del sistema:



**Interfaz Externa:** El diseño de la interfaz visual debe ser amigable para los usuarios que interactuarán con la aplicación, de forma que les facilite el entendimiento de las funcionalidades que el mismo brinda, además de poseer colores amigables y refrescantes para una mejor interacción entre el usuario y la aplicación. La interfaz constará de un grupo de botones, editores para textos, desplegables e informaciones que permitirán al usuario un fácil manejo de la aplicación y una realización rápida de las actividades. Toda la información referente a los textos, mensajes y opciones serán mostrados en idioma español. El diseño de la interfaz permitirá mostrar mensajes para la guía de usuarios en caso de errores en entradas inválidas de los datos o de confirmación de realización de actividades.

**Usabilidad:** El sistema estará diseñado de manera que los usuarios adquieran las habilidades necesarias para explotarlo en un tiempo reducido.

**Seguridad:** Se garantizará que la aplicación sea utilizada correctamente por cada usuario según sus niveles permitidos, de acuerdo a los principios de confidencialidad, disponibilidad e integridad.

**Soporte:** Se tendrá un manual de ayuda que le facilite al usuario el trabajo y la navegación por la herramienta.

**Respaldo y recuperación de base de datos:** El sistema debe permitir la realización de copias de respaldo hacia otros dispositivos de almacenamiento para prevenir pérdida de información. Esta base de datos debe ser posible de recuperar mediante las copias de respaldo previamente creadas.

### **Software:**

En el cliente:

- Navegador: Mozilla Firefox o Internet Explorer 8

### **Hardware:**

En el servidor:

- Procesador Intel Pentium III o superior.
- 2 Gb de memoria RAM.
- Disco duro de 80 Gb o más.
- Monitor tipo VGA o superior.



- Tarjeta de Red.

En el cliente:

- Procesador Intel Pentium IV o superior.
- 512 Mb de memoria RAM.
- Monitor tipo VGA o superior.
- Tarjeta de Red.

## 2.6. Modelo de Casos de Uso

### 2.6.1. Actores del Sistema

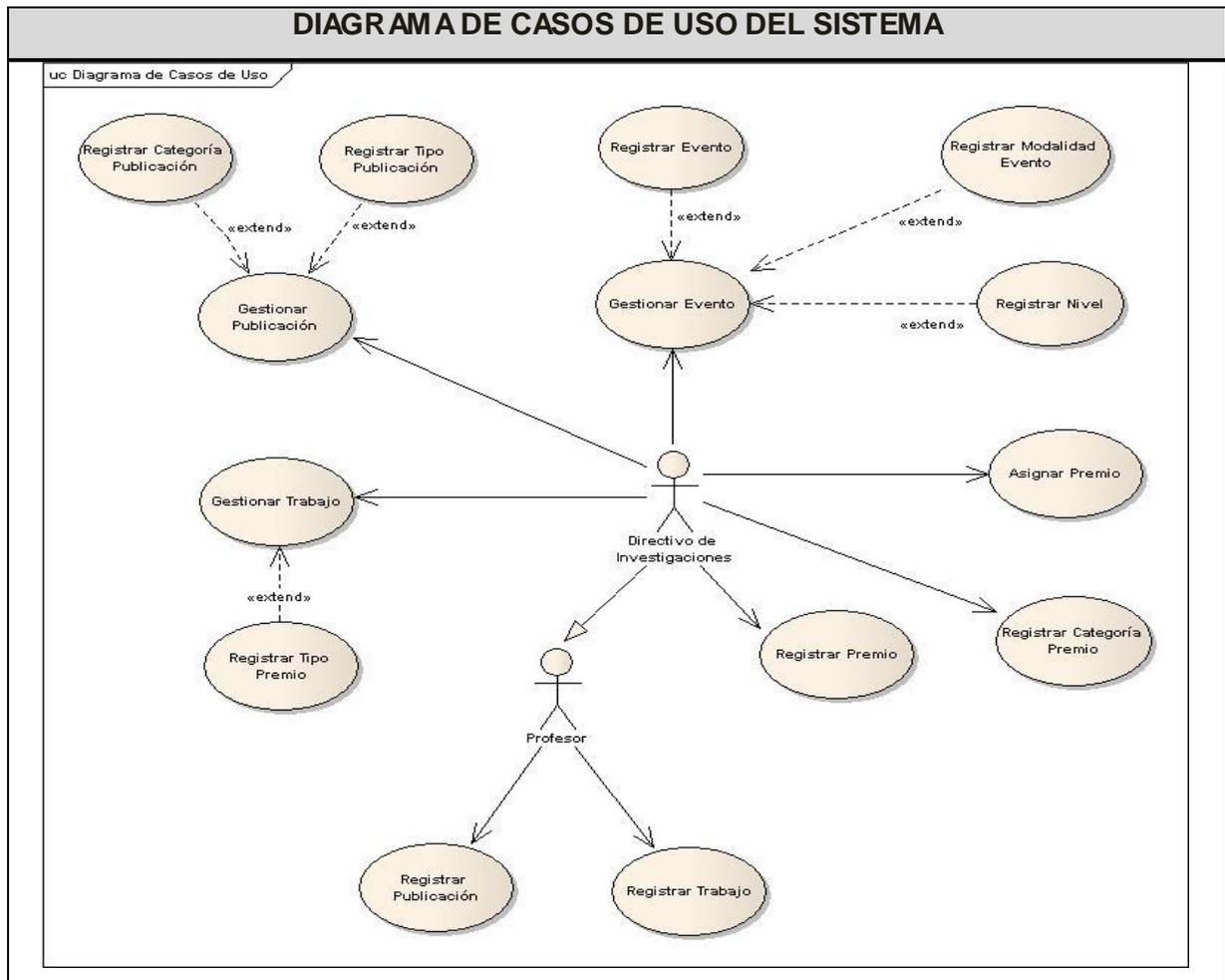
Un actor del sistema es una entidad externa, representando el rol de una o varias personas, un equipo o un sistema automatizado que interactúa con el sistema, por lo que no forma parte del mismo. Puede intercambiar información o ser un recipiente pasivo de información.

Actor	Función
Profesor	Es el encargado de registrar trabajos y publicaciones.
Directivo de Investigaciones	Es el encargado de gestionar los trabajos, las publicaciones y los premios que se le puede asignar a un profesor o a un estudiante. Podrá registrar eventos, modificarlos o eliminarlos, gestionar reportes y asignar premios. Puede ser el Vicedecano de Investigación, el Asesor de Investigaciones y o el Subdirector de Investigaciones y Postgrado del CESIM.



**2.6.2. Diagrama de Casos de Uso del Sistema**

Los diagramas de casos de uso explican gráficamente un conjunto de casos de uso del sistema. Tiene como objeto mostrar una clase de diagrama contextual que permite conocer rápidamente los actores externos de un sistema y las formas básicas en que lo utilizan.



**2.6.3. Especificación de Casos de Usos del Sistema**

A continuación se describen las especificaciones de los casos de uso que forman parte del sistema.



CU Gestionar Evento

<b>Caso de Uso</b>	CU Gestionar Evento
<b>Usuario</b>	Directivo de Investigaciones
<b>Descripción</b>	El caso de uso inicia cuando un Directivo de Investigaciones decide realizar acciones como buscar, registrar, modificar o eliminar un evento.
<b>Referencia</b>	RF 1

CU Registrar Evento

<b>Caso de Uso</b>	CU Registrar Evento
<b>Usuario</b>	Directivo de Investigaciones
<b>Descripción</b>	El caso de uso inicia cuando un usuario desea registrar un nuevo evento.
<b>Referencia</b>	RF 2

CU Registrar Modalidad Evento

<b>Caso de Uso</b>	CU Registrar Modalidad Evento
<b>Usuario</b>	Directivo de Investigaciones
<b>Descripción</b>	El caso de uso inicia cuando un Directivo de Investigaciones decide registrar una modalidad de un evento.
<b>Referencia</b>	RF 3

CU Buscar Modalidad Evento

<b>Caso de Uso</b>	CU Buscar Modalidad Evento
<b>Usuario</b>	Directivo de Investigaciones
<b>Descripción</b>	El caso de uso inicia cuando un Directivo de Investigaciones decide buscar o eliminar una modalidad de un evento
<b>Referencia</b>	RF 4



CU Registrar Nivel Evento

<b>Caso de Uso</b>	CU Registrar Nivel Evento
<b>Usuario</b>	Directivo de Investigaciones
<b>Descripción</b>	El caso de uso inicia cuando un Directivo de Investigaciones decide registrar o eliminar un nivel de evento.
<b>Referencia</b>	RF 5

CU Gestionar Trabajo

<b>Caso de Uso</b>	CU Gestionar Trabajo
<b>Usuario</b>	Directivo de Investigaciones
<b>Descripción</b>	El caso de uso inicia cuando un Directivo de Investigaciones decide realizar acciones como buscar, modificar, eliminar o aprobar un trabajo.
<b>Referencia</b>	RF 6

CU Registrar Tipo Premio

<b>Caso de Uso</b>	CU Registrar Tipo Premio
<b>Usuario</b>	Directivo de Investigaciones
<b>Descripción</b>	El caso de uso inicia cuando un Directivo de Investigaciones decide registrar o eliminar un premio que puede ser asignado a un trabajo.
<b>Referencia</b>	RF 7

CU Gestionar Publicación

<b>Caso de Uso</b>	CU Gestionar Publicación
<b>Usuario</b>	Directivo de Investigaciones
<b>Descripción</b>	El caso de uso inicia cuando un Directivo de Investigaciones decide realizar acciones como buscar, modificar, eliminar o aprobar una publicación.
<b>Referencia</b>	RF 8



CU Registrar Tipo Publicación

<b>Caso de Uso</b>	CU Registrar Tipo Publicación
<b>Usuario</b>	Directivo de Investigaciones
<b>Descripción</b>	El caso de uso inicia cuando un Directivo de Investigaciones decide registrar o eliminar un tipo de publicación.
<b>Referencia</b>	RF 9

CU Registrar Categoría Publicación

<b>Caso de Uso</b>	CU Registrar Categoría Publicación
<b>Usuario</b>	Directivo de Investigaciones
<b>Descripción</b>	El caso de uso inicia cuando un Directivo de Investigaciones decide registrar o eliminar una categoría de publicación.
<b>Referencia</b>	RF 10

CU Registrar Premio

<b>Caso de Uso</b>	CU Registrar Premio
<b>Usuario</b>	Directivo de Investigaciones
<b>Descripción</b>	El caso de uso inicia cuando un Directivo de Investigaciones decide realizar acciones como registrar o eliminar un premio, que puede ser asignado a un profesor.
<b>Referencia</b>	RF 11

CU Registrar Categoría Premio

<b>Caso de Uso</b>	CU Registrar Categoría Premio
<b>Usuario</b>	Directivo de Investigaciones
<b>Descripción</b>	El caso de uso inicia cuando un Directivo de Investigaciones decide registrar o eliminar una categoría de premio.
<b>Referencia</b>	RF 12



CU Buscar Persona

<b>Caso de Uso</b>	CU Buscar Persona
<b>Usuario</b>	Usuario
<b>Descripción</b>	El caso de uso se inicia cuando se solicita buscar un profesor para asignarle un premio.
<b>Referencia</b>	RF 13

CU Asignar Premio

<b>Caso de Uso</b>	CU Asignar Premio
<b>Usuario</b>	Directivo de Investigaciones
<b>Descripción</b>	El caso de uso inicia cuando un Directivo de Investigaciones decide asignarle un premio a un determinado profesor.
<b>Referencia</b>	RF 14

CU Registrar Publicación

<b>Caso de Uso</b>	CU Registrar Publicación
<b>Usuario</b>	Profesor
<b>Descripción</b>	El caso de uso inicia cuando un profesor desea registrar una nueva publicación.
<b>Referencia</b>	RF 15

CU Registrar Trabajo

<b>Caso de Uso</b>	CU Registrar Trabajo
<b>Usuario</b>	Profesor
<b>Descripción</b>	El caso de uso inicia cuando un profesor desea registrar un nuevo trabajo.
<b>Referencia</b>	RF 16



**2.6.4. Casos de uso expandidos**

Para conocer en detalle lo que trae asociado cada caso de uso, se muestra la descripción general de dos de los CU más explicativos, los restantes se pueden encontrar en el Anexo1.

<b>CASO DE USO:</b>	Asignar Premio	
<b>Resumen:</b>	El caso de uso inicia cuando el Directivo desea realizar una asignación de un premio a un profesor determinado. El caso de uso concluye cuando se termina la asignación del premio.	
<b>Actores:</b>	Directivo de Investigaciones.	
<b>Precondiciones:</b>	Para asignar un premio, debe haber sido seleccionado un profesor y debe existir al menos un premio registrado en el sistema.	
<b>REFERENCIAS</b>		
<b>Referencias:</b>		
<b>FLUJO NORMAL DE EVENTOS</b>		
<b>Acción del Actor</b>	<b>Respuesta del Sistema</b>	
1. El caso de uso inicia cuando el actor accede a la opción "Asignar Premio".	2. Muestra los datos de la persona seleccionada en una nueva ventana: <ul style="list-style-type: none"> <li>▪ Nombre del Profesor</li> </ul> Permitiendo seleccionar los datos: <ul style="list-style-type: none"> <li>▪ Premio</li> <li>▪ Categoría</li> <li>▪ Certificación</li> </ul> Muestra también el Listado de premios obtenidos por ese profesor y por cada premio brinda las opciones: <ul style="list-style-type: none"> <li>▪ Ver Certificación del premio seleccionado. (Botón "Ver")</li> <li>▪ Eliminar el premio. (Botón "Eliminar")</li> </ul> Brinda además las opciones: <ul style="list-style-type: none"> <li>▪ Asignarle un nuevo premio a la persona. (Botón "Asignar")</li> <li>▪ Subir la certificación del premio. (Botón "Subir",</li> </ul>	



	<p>ver Caso de Uso: Subir Certificación)</p> <ul style="list-style-type: none"> <li>▪ Cancelar la asignación. (Botón “Cancelar”)</li> </ul>
<p>3. Decide realizar una de las siguientes acciones:</p> <ul style="list-style-type: none"> <li>▪ Asignarle un nuevo premio a la persona.</li> <li>▪ Cancelar la asignación, ver Flujo Alterno 1.</li> <li>▪ Ver certificación, Ver Flujo Alterno 2.</li> <li>▪ Eliminar un premio asignado. (Ver Sección “Eliminar Asignación”)</li> </ul>	<p>4. Si el usuario selecciona el botón “Asignar”:</p> <p>Verifica que los datos insertados están completos y son correctos.</p> <ul style="list-style-type: none"> <li>▪ Detecta que existen campos vacíos que son obligatorios. (Ver Flujo Alterno 3)</li> <li>▪ Detecta errores en los datos registrados. (Ver Flujo Alterno 4)</li> <li>▪ Detecta que ya existe ese premio asignado a esa persona. (Ver Flujo Alterno 5)</li> <li>▪ Se asigna el premio y se actualiza la lista de premios asignados del profesor seleccionado..</li> </ul>
<b>Flujo Alterno 1</b>	
	<p>4. El premio no es registrado.</p> <p>5. Vuelve a la vista “Buscar Profesor”.</p>
<b>Flujo Alterno 2</b>	
	<p>4. Muestra en una nueva ventana donde se encuentra la imagen de la certificación de la publicación.</p>
<b>Flujo Alterno 3</b>	
	<p>5. Muestra el mensaje: “<i>El campo es obligatorio.</i>” por cada uno de los campos obligatorios.</p> <p>6. Permite volver a registrar los datos.</p>
<b>Flujo Alterno 4</b>	
	<p>5. Muestra un mensaje de error por cada campo registrado con error.</p> <p>6. Permite volver a registrar los datos.</p>
<b>Flujo Alterno 5</b>	
	<p>4. Muestra el mensaje: “<i>El premio ya fue asignado a este profesor.</i>”.</p> <p>5. Permite volver a registrar los datos.</p>
<b>Sección: “Eliminar Asignación”</b>	
<b>Acción del Actor</b>	<b>Respuesta del Sistema</b>



	<p>1. Muestra el mensaje: “¿Está seguro que desea eliminar la asignación?”.</p> <ul style="list-style-type: none"> <li>▪ Si el usuario selecciona el botón “Aceptar” del mensaje de confirmación, se elimina el premio.</li> <li>▪ Si el usuario selecciona el botón “Cancelar” del mensaje de confirmación, ver Flujo Alterno 1.</li> </ul>
<b>Flujo Alterno 1</b>	
	2. El premio no es eliminado.

<b>CASO DE USO:</b>	Registrar Trabajo
<b>Resumen:</b>	El caso de uso inicia cuando el actor desea registrar los datos de un trabajo determinado. El caso de uso concluye cuando se termina el registro del trabajo.
<b>Actores:</b>	Profesor
<b>Precondiciones:</b>	-
<b>REFERENCIAS</b>	
<b>Referencias:</b>	
<b>FLUJO NORMAL DE EVENTOS</b>	
<b>Acción del Actor</b>	<b>Respuesta del Sistema</b>
1. El caso de uso inicia cuando el actor accede a la opción “Registrar Trabajo”.	<p>2. Muestra los campos para el registro de los datos necesarios para crear un trabajo:</p> <ul style="list-style-type: none"> <li>▪ Autores</li> <li>▪ Co-Autores</li> <li>▪ Título</li> <li>▪ Premio</li> <li>▪ Expuesto</li> <li>▪ Año</li> <li>▪ Evento</li> <li>▪ Certificación</li> </ul>
<p>3. Inserta los datos especificados.</p> <p>4. Decide realizar una de las siguientes acciones:</p> <ul style="list-style-type: none"> <li>▪ Registrar el trabajo en el sistema.</li> </ul>	<p>5. Si el usuario selecciona el botón “Registrar”:</p> <p>Verifica que los datos insertados están completos y son correctos.</p> <ul style="list-style-type: none"> <li>▪ Detecta que existen campos vacíos que son</li> </ul>



<p>(Botón “Registrar”)</p> <ul style="list-style-type: none"> <li>▪ Cancelar el registro (Botón “Cancelar”, ver Flujo Alternativo 1)</li> </ul>	<p>obligatorios de llenar. (Ver Flujo Alternativo 2)</p> <ul style="list-style-type: none"> <li>▪ Detecta errores en los datos registrados. (Ver Flujo Alternativo 3)</li> <li>▪ Detecta que ya existe ese trabajo. (Ver Flujo Alternativo 4)</li> <li>▪ Se insertan los datos del trabajo.</li> </ul>
<b>Flujo Alternativo 1</b>	
	<ol style="list-style-type: none"> <li>5. El trabajo no es registrado.</li> <li>6. Regresa a la vista “Buscar Trabajo”.</li> </ol>
<b>Flujo Alternativo 2</b>	
	<ol style="list-style-type: none"> <li>6. Muestra el mensaje: <i>“El campo es obligatorio.”</i> por cada uno de los campos obligatorios.</li> <li>7. Permite volver a registrar los datos.</li> </ol>
<b>Flujo Alternativo 3</b>	
	<ol style="list-style-type: none"> <li>6. Muestra un mensaje de error por cada campo registrado con error.</li> <li>7. Permite volver a registrar los datos.</li> </ol>
<b>Flujo Alternativo 4</b>	
	<ol style="list-style-type: none"> <li>6. Muestra el mensaje: <i>“El trabajo ya está registrado.”</i>.</li> <li>7. Permite volver a registrar un nuevo trabajo.</li> </ol>

En el presente capítulo se ha realizado un análisis de cómo ocurre el flujo de la información entre la Dirección de Investigaciones y los profesores de la facultad. De la información obtenida, se definieron las necesidades de funcionamiento de la aplicación a implementar. A partir de los requerimientos funcionales y no funcionales, se definieron los Casos de Usos del Sistema, que son las funcionalidades que se van a implementar y se describió además, cada uno de los Casos de Usos del Sistema.



Este capítulo tiene el objetivo de plantear la concepción general del diseño del sistema propuesto y cómo se implementa este. Se presentan los diagramas de clases de diseño web que detallan la interacción de las distintas páginas; se estructura la información que se desea que persista a través del diseño de la base de datos y se describe la distribución física del sistema en términos de cómo se distribuye la funcionalidad entre los nodos de cómputo.

### 3.1. Descripción de la arquitectura. Fundamentación

En el diseño de sistemas informáticos en la actualidad se suele usar las arquitecturas multinivel o programación por capas. La programación por capas es un estilo en el que el objetivo primordial es separar la lógica del negocio de la lógica del diseño; un ejemplo básico de esto consiste en separar la capa de datos de la capa de presentación al usuario. En dichas arquitecturas a cada nivel se le confía una misión simple. La ventaja principal de este estilo es que el desarrollo se puede llevar a cabo en varios niveles y, en caso de que sobrevenga algún cambio, sólo se ataca al nivel requerido sin tener que revisar entre código mezclado.

Las aplicaciones web pueden desarrollarse utilizando cualquier arquitectura. La arquitectura del patrón Modelo-Vista-Controlador (MVC) es un paradigma de programación bien conocido para el desarrollo de aplicaciones con interfaz gráfica.

El patrón MVC es un patrón de arquitectura de software en el cual todo el proceso está dividido en tres capas, típicamente estas capas son el Modelo, la Vista y el Controlador. El Modelo incorpora la capa de dominio y persistencia, es el encargado de guardar los datos en un medio persistente (ya sea una base de datos, un archivo de texto, etc.). Se refiere a la lógica del negocio o servicio y los datos asociados con la aplicación.



La Vista se encarga de presentar la interfaz al usuario, en sistemas Web, esto es típicamente HTML, aunque pueden existir otros tipos de vistas. En ella sólo se deben realizar operaciones simples. Es la encargada de la presentación de los datos.

El Controlador es el que escucha los cambios en la vista y los envía al modelo, este último retorna los datos a la vista. Es el que atiende las peticiones y componentes para la toma de decisiones de la aplicación. El propósito del MVC es aislar los cambios. Es una arquitectura preparada para los cambios, que separa los datos y lógica de negocio de la lógica de presentación.

Django es un framework de desarrollo web que sigue el modelo MVC, aunque los desarrolladores del proyecto prefieren llamarlo Modelo-Plantilla-Vista (*Model-Template-View* "MVT"). El controlador en este caso sería el propio *framework*. Separa la programación del diseño, con lo que se puede mantener trabajando en paralelo a programadores y diseñadores, ahorrando mucho tiempo. Esto también permite cambiar en cualquier momento el diseño del sitio, sin preocuparse porque la aplicación pueda resultar dañada.

La principal ventaja en lo referente al diseño de usar Django, es que directamente obliga a diseñar el servidor de una determinada manera forzando a cumplir con el patrón MVC. Una aplicación MVC en Django estará distribuida de la siguiente forma:

- El archivo *models.py* contiene una descripción de la tabla de la base de datos, la declaración de la base de datos se hace de la misma forma que si se declarase una clase Python. El contenido de este fichero se corresponderá con el modelo descrito en el patrón MVC. Usando esta clase se pueden crear, buscar, actualizar y borrar entradas de la base de datos usando código Python sencillo en lugar de escribir declaraciones SQL. Será el propio framework el que se encargará de realizar la asociación entre los objetos declarados y sus tablas correspondientes en la base de datos.
- El archivo *urls.py* define patrones de URLs asociados a un manejador. Cuando recibe una petición comparará la URL solicitada con el patrón definido y decide a qué manejador redirigir la petición. Se corresponderá con el controlador del patrón MVC.
- El archivo *views.py* contiene los manejadores a los que redirige la petición el fichero "urls.py". Cuando el manejador es llamado, atiende la petición y devuelve la respuesta en el formato esperado por el cliente. Se corresponde con la vista del patrón MVC.



Tomadas en su conjunto, estas piezas se aproximan al patrón de diseño MVC. Dicho de manera más fácil, MVC precisa una forma de desarrollar software en la que el código para definir y acceder a los datos (el modelo) está separado de la redirección lógica de asignación de ruta (el controlador), que a su vez está separado de la interfaz del usuario o datos devueltos (la vista).

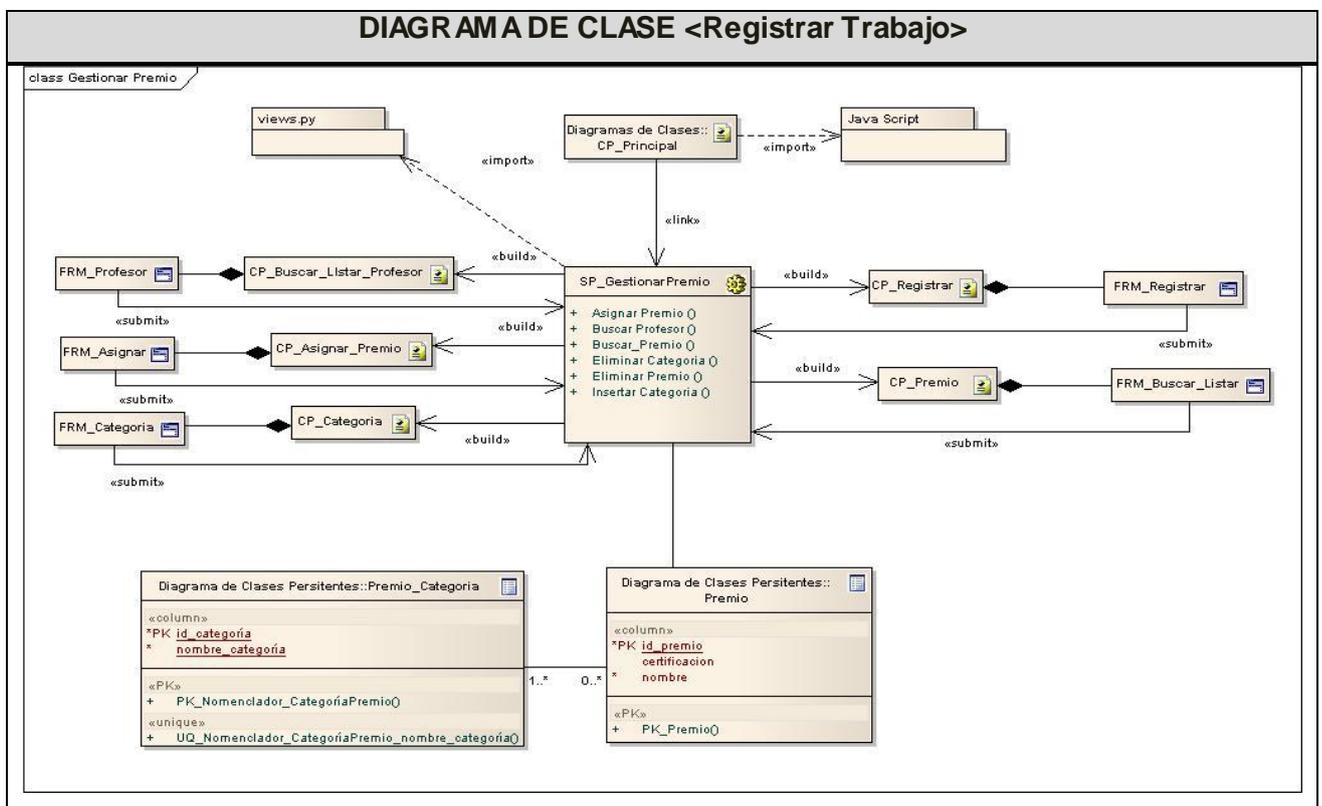
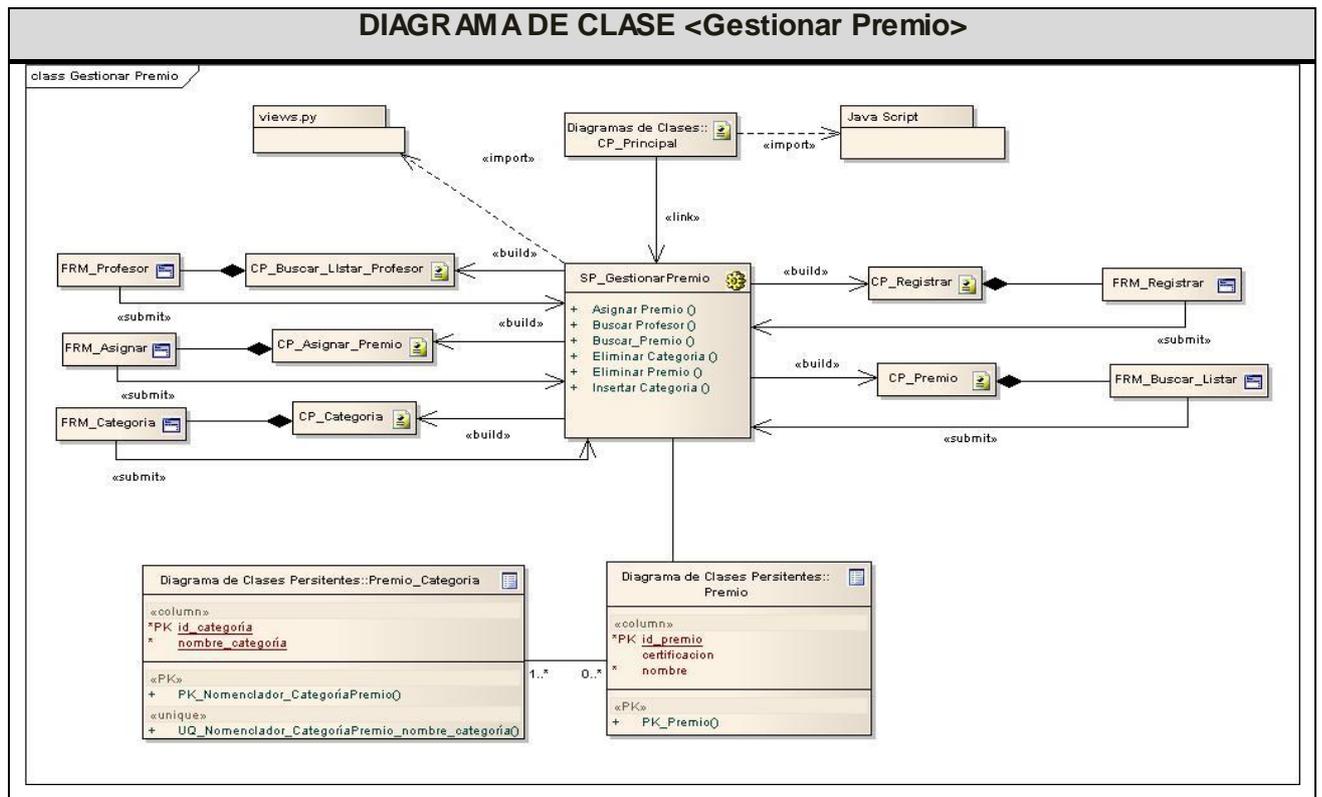
### **3.2. Principios de diseño**

El diseño es la antesala a la implementación del sistema, mediante el Modelo de Diseño se puede verificar el cumplimiento de los requerimientos del sistema sin tener un prototipo funcional de este, además define e identifica las consecuencias del ambiente de implementación; es un modelo o guía para realizar la implementación del sistema que se va a desarrollar.

#### **3.2.1. Diagrama de clases del diseño**

Un diagrama de clases presenta las clases del sistema con sus relaciones estructurales y de herencia, todo el código que irá creando las páginas, así como el contenido dinámico de estas una vez que estén en el navegador del cliente. En el caso de las aplicaciones Web, el diagrama de clases representa las colaboraciones que ocurren entre las páginas, donde cada página lógica puede ser representada como una clase. Es muy importante pues estos son los artefactos que se necesitan modelar para que el desarrollador los implemente y obtener así el producto final con la calidad requerida.

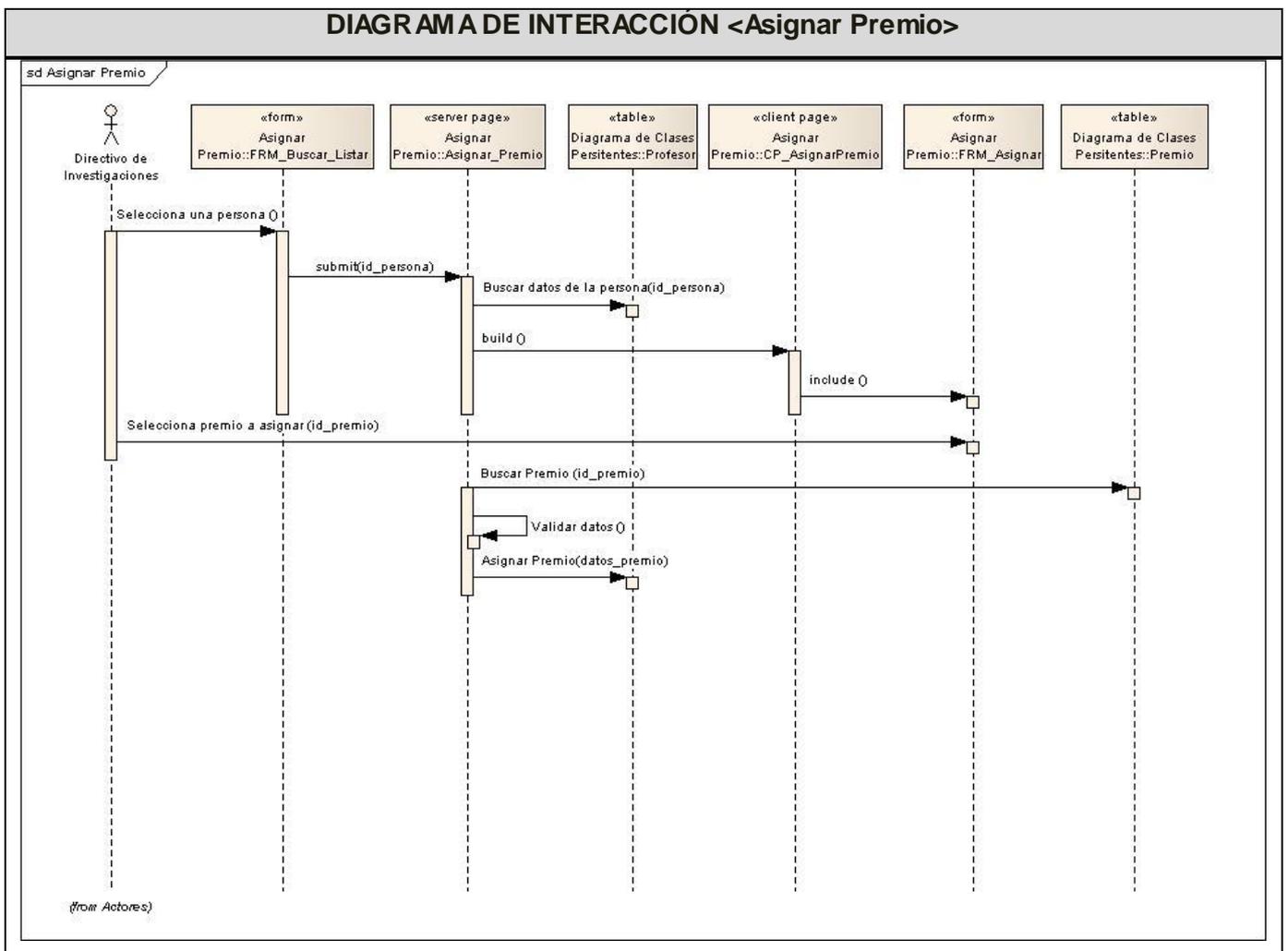
El diagrama de clases web, fue definido, a partir de los diferentes casos de uso del sistema. A continuación se muestran los diagramas de clases para los Casos de Uso cuyas descripciones generales se encuentran en el capítulo anterior. En el Anexo 2 se encuentra una representación más abarcadora de los restantes diagramas.

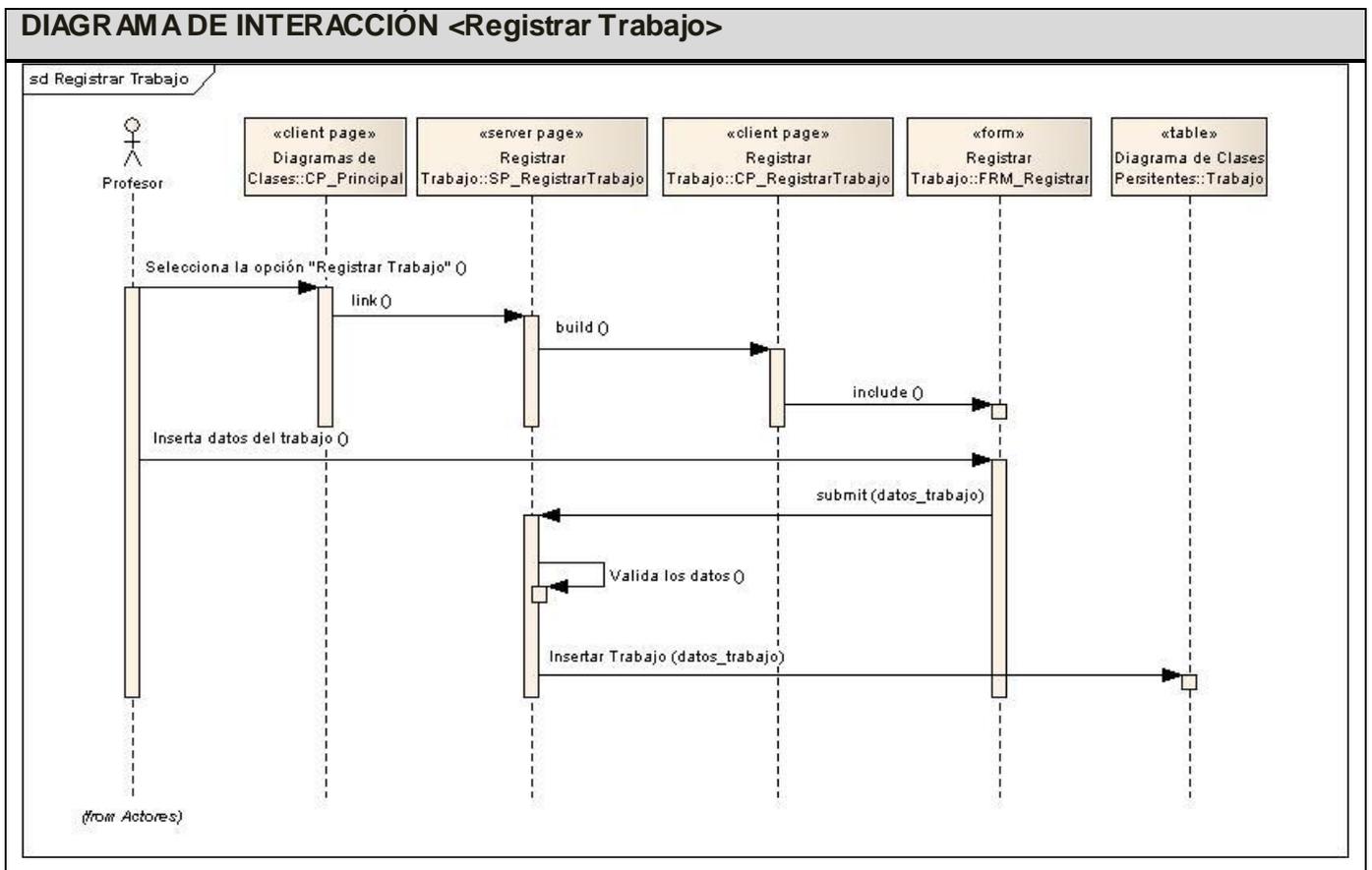
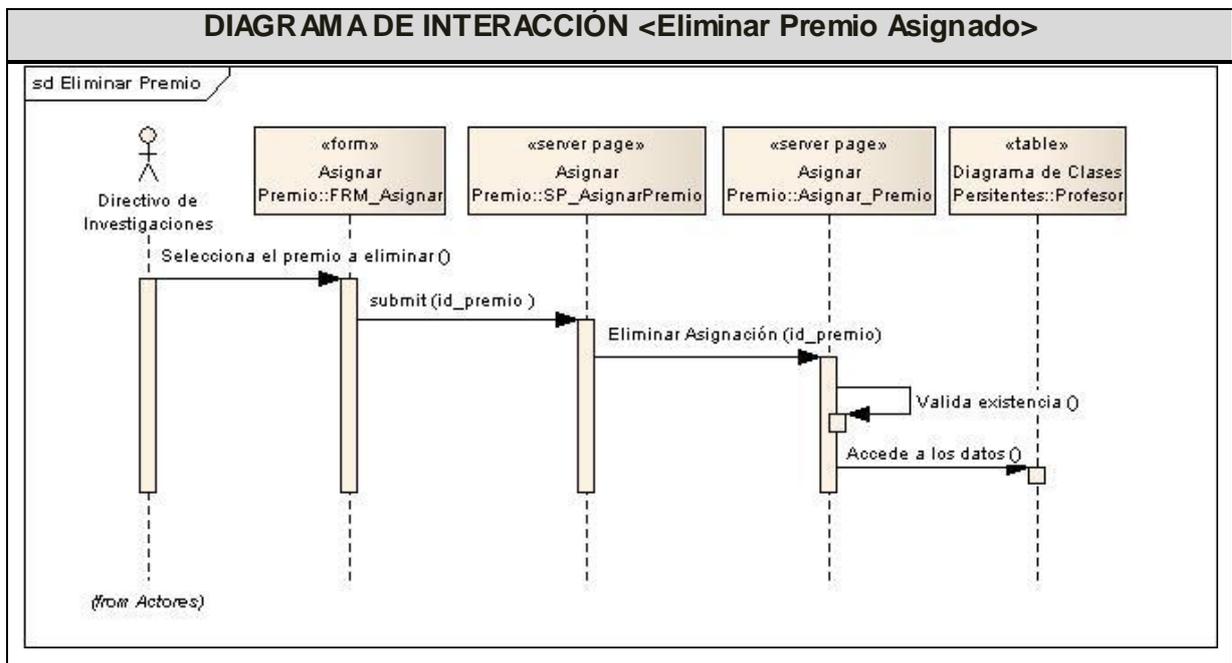




### 3.2.2. Diagramas de Interacción en el diseño

Los Diagramas de Interacción pertenecen al grupo de Diagramas de Comportamiento, y se clasifican en Diagramas de Secuencia o Colaboración. Específicamente el Diagrama de Secuencia consiste en resaltar el ordenamiento temporal de los mensajes, presenta un conjunto de objetos con los mensajes que reciben y que envían entre ellos. Los Diagramas de Interacción son de gran importancia en el diseño de un sistema, debido a que permiten observar las interacciones que ocurren entre los distintos objetos que participan en un escenario determinado. A continuación se muestra una selección de los diagramas de interacción que tienen estrecha relación con los diagramas de clases anteriormente representados. Los restantes se presentan en el Anexo 3.







### 3.2.3. Descripción de las clases del diseño

A continuación se presentan las descripciones de las clases de diseño que tienen relación con las funcionalidades presentadas, las restantes se podrán encontrar en el Anexo 4.

#### Descripción de las clases para Asignar Premio

<b>NOMBRE:</b>	Buscar_Listar_Profesor	
<b>Tipo de Clase:</b>	Interfaz	
<b>Atributo</b>	<b>Tipo</b>	
Nombre	Campo de texto	
Usuario	Campo de texto	
Buscar	Botón	
Cancelar	Botón	
Listado de Profesores	Tabla	
Asignar Premio	Vínculo	
<b>Descripción:</b>	Esta clase muestra una interfaz para buscar un profesor. Además muestra el resultado de la búsqueda de los mismos.	

<b>NOMBRE:</b>	Asignar_Premio_Profesor	
<b>Tipo de Clase:</b>	Interfaz	
<b>Atributo</b>	<b>Tipo</b>	
Profesor	Label	
Premio	Lista / Menú	
Categoría	Lista / Menú	
Certificación	Botón	
Asignar	Botón	
Cancelar	Botón	
Listado de Premios	Tabla	
Ver Certificación	Vínculo	
Eliminar	Vínculo	
<b>Descripción:</b>	Esta clase muestra una interfaz para asignarle un premio a un profesor. Además muestra los premios que ya pertenecen a dicho profesor.	



### Descripción de las clases para Registrar Trabajo

<b>NOMBRE:</b>	Registrar_Trabajo	
<b>Tipo de Clase:</b>	Interfaz	
<b>Atributo</b>	<b>Tipo</b>	
Título	Campo de texto	
Año	Campo de texto	
Expuesto	Casilla de Verificación	
Evento	Lista/ Menú	
Premio	Lista/ Menú	
Subir Certificación	Botón	
Listado de Autores	Tabla	
Listado de Coautores	Tabla	
Buscar Autores	Vínculo	
Buscar Coautores	Vínculo	
Registrar	Botón	
Cancelar	Botón	
<b>Descripción:</b>	Esta clase muestra una interfaz para registrar todos los datos referentes a un trabajo.	

En el presente capítulo se logró presentar la descripción del diseño a través de diagramas de clases del diseño de la aplicación. Para la realización de estos diagramas se tuvo en cuenta la utilización de estereotipos web, que describen la relación entre las páginas. Por lo que se obtuvo además, el diagrama de clases persistentes, para generar a partir del mismo la base de datos. Se definieron los principios de diseño, abordando la descripción de la arquitectura definida, patrones y estándares a utilizar.



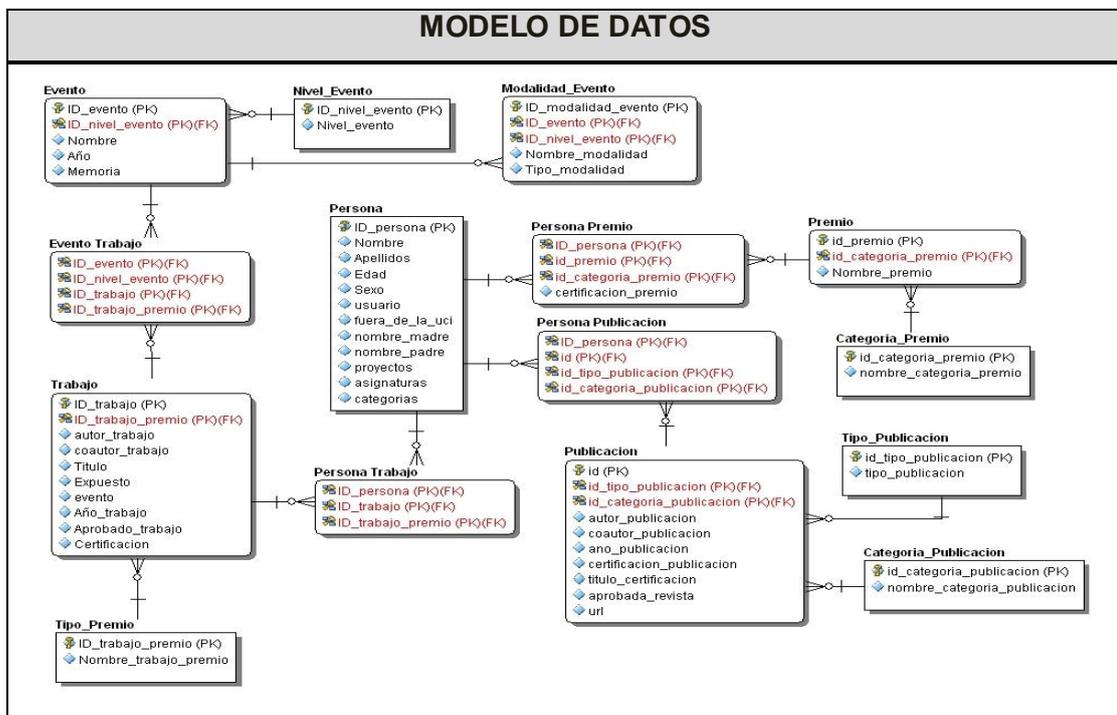
CAPÍTULO 4  
IMPLEMENTACIÓN Y PRUEBA

En el presente capítulo se realiza el flujo de trabajo Implementación definido por la metodología de desarrollo RUP y se presentan el modelo de datos obtenido y la descripción de las tablas que forman parte de este. Así como los principales artefactos relacionados con este flujo de trabajo: los diagramas de despliegue y de componentes.

3.1. Modelo de Datos

El modelo de datos permite describir los elementos de la realidad que intervienen en un problema dado y la forma en que se relacionan esos elementos entre sí. Constituye un artefacto de gran importancia dentro del proceso que incluye generar la base de datos en la metodología RUP. El mismo proporciona una representación visual y física de los datos persistentes del sistema, que finalmente formarán las tablas de la base de datos.

En este epígrafe se representa el diseño de la base de datos a través del modelo de datos.





### 3.1.1. Descripción de las tablas

La descripción de las tablas que forman parte del modelo de datos permite comprender los diagramas que son resultado del diseño de la base de datos de la aplicación. En estas descripciones se podrá encontrar información más específica de las clases de más relevancia, los nomencladores y las restantes tablas de menor importancia se podrán encontrar en el Anexo 5.

<b>NOMBRE:</b> Evento		
<b>Descripción:</b> Almacena los datos correspondientes a los eventos.		
<b>Atributo</b>	<b>Tipo</b>	<b>Descripción</b>
id_evento	IntegerField	Llave primaria: representa el identificador de un evento. Atributo que no cambia su valor y mantiene la integridad y relación entre los datos.
nombre	CharField	Contiene el nombre del evento.
año	DateField	Contiene el año en que se realizó el evento.
id_nivel_evento	IntegerField	Llave foránea: contiene los id de los niveles que puede tener un evento.
memorias	BooleanField	Contiene el valor de "True" si el evento posee memorias, "False" en caso contrario.

<b>NOMBRE:</b> Publicación		
<b>Descripción:</b> Almacena los datos correspondientes a las publicaciones que realizan los profesores.		
<b>Atributo</b>	<b>Tipo</b>	<b>Descripción</b>
id_publicacion	IntegerField	Llave primaria: representa el identificador de una publicación. Atributo que no cambia su valor y mantiene la integridad y relación entre los datos.
id_tipo_publicación	IntegerField	Llave foránea: contiene los id de los tipos de publicaciones.



id_categoria_publicación	IntegerField	Llave foránea: contiene los id de las categorías de las publicaciones.
autor	CharField	Contiene los nombres de los autores de una publicación.
año	DateField	Contiene el año en que se publicó el artículo.
certificación	ImageField	Contiene la imagen de la certificación de la publicación, la misma valida la veracidad de la publicación.
título	CharField	Contiene el título de la publicación.
URL	CharField	Contiene el lugar donde se publicó, si es en una revista referenciada o en un sitio web, entre otros.
Aprobada_publicación	BooleanField	Contiene el valor de "True" si la publicación fue aprobada, "False" en caso contrario.

<b>NOMBRE:</b> Trabajo		
<b>Descripción:</b> Almacena los datos correspondientes a los trabajos presentados en eventos.		
Atributo	Tipo	Descripción
id_trabajo	IntegerField	Llave primaria: representa el identificador de un trabajo. Atributo que no cambia su valor y mantiene la integridad y relación entre los datos.
id_tipo_premio	CharField	Llave foránea: contiene el id de los premios que obtuvo el trabajo.
evento	CharField	Contiene el nombre del evento en el que se presentó el trabajo.
Autor	CharField	Contiene los nombres de los autores trabajo.
titulo	CharField	Contiene el título del trabajo.
año	DateField	Contiene el año en el que se presentó el



		trabajo.
expuesto	BooleanField	Contiene "True" si el trabajo fue expuesto, "False" en caso contrario.
obtuvo_premio	BooleanField	Contiene "True" si el trabajo alcanzó algún premio, "False" en caso contrario.
aprobado_trabajo	BooleanField	Contiene "True" si el trabajo fue aprobado, "False" en caso contrario.
certificación	ImageField	Contiene la imagen de la certificación del trabajo, la misma valida la veracidad del premio alcanzado por el mismo o de la participación en el evento donde se presentó.

<b>NOMBRE:</b> Premio_Persona		
<b>Descripción:</b> Almacena los datos correspondientes a los premios asignados a un profesor.		
Atributo	Tipo	Descripción
id_premio	IntegerField	Llave primaria: representa el identificador de un premio. Atributo que no cambia su valor y mantiene la integridad y relación entre los datos.
id_persona	CharField	Llave foránea: contiene el id de la persona a la cual se le va a signar un premio.
certificación	ImageField	Contiene la imagen de la certificación del trabajo, la misma valida la veracidad del premio alcanzado por el mismo o de la participación en el evento donde se presentó.

### 3.2. Implementación

La implementación es la fase más esperada en un proceso de desarrollo de un producto software, es donde se hacen realidad todas las ideas y artefactos que han sido modelados por el equipo de

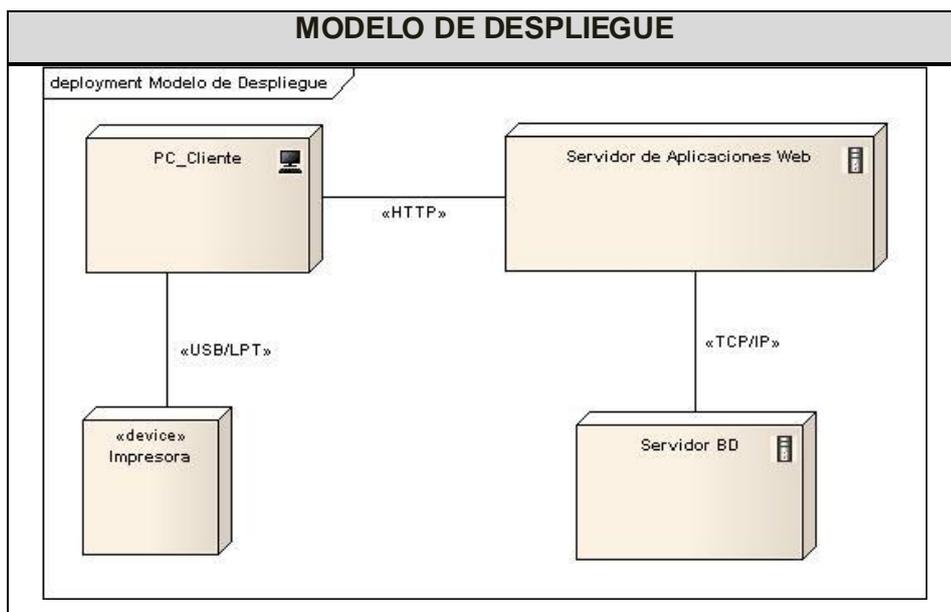


trabajo responsable de la solución; no constituye una etapa independiente y formalmente delimitada en el proceso, ya que la metodología de desarrollo utilizada permite que partes de la solución que conceptualmente estaban definidas y que se podrían ir prototipando de forma funcional se le diera luz verde para implementación mucho antes de que estos artefactos fueran modelados.

### 3.2.1. Diagrama de despliegue

El modelo de despliegue contiene los nodos que conforman la topología de hardware sobre la que se ejecuta el sistema. Muestra las relaciones entre el hardware y el software en el sistema final. Se representa como un grafo de nodos unidos por conexiones de comunicación. El diagrama de despliegue es un modelo de objetos que describe la distribución física del sistema en términos de cómo se distribuye la funcionalidad entre los nodos de cómputo. Es una colección de nodos y arcos; donde cada nodo representa un recurso de cómputo, normalmente un procesador o un dispositivo de hardware similar.

Muestra la configuración de los componentes hardware, los procesos, los elementos de procesamiento y los objetos que existen en tiempo de ejecución. En este tipo de diagramas intervienen nodos, asociaciones de comunicación, componentes dentro de los nodos y objetos que se encuentran a su vez dentro de los componentes. Un nodo es un objeto físico, es decir una máquina que se compone habitualmente de por lo menos, memoria y capacidad de procesamiento, a su vez puede estar formada por otros componentes.

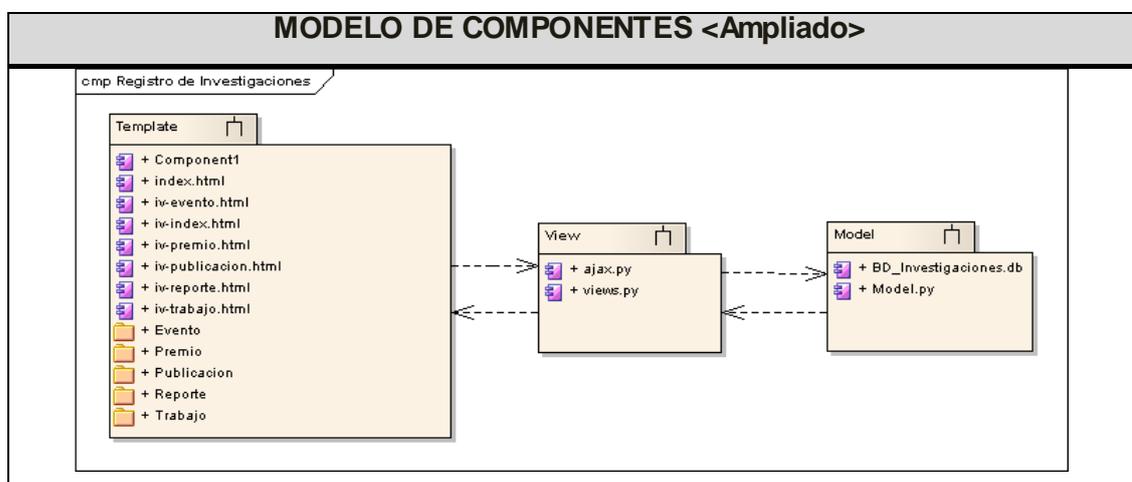
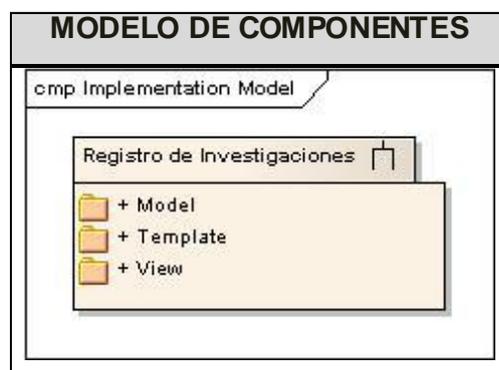


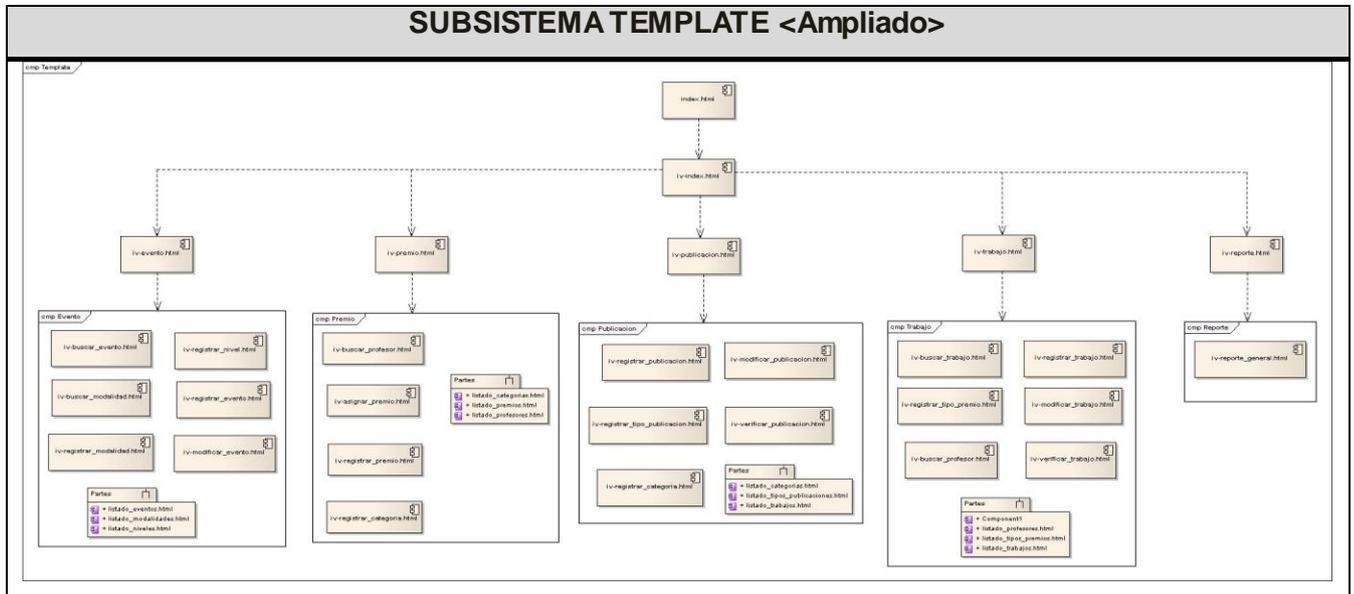


El modelo de despliegue que se propone está compuesto por tres computadoras, una de ellas haciendo función de servidor de Base de Datos utilizando como SGBD Postgres 8.2, otra máquina como PC cliente, la cual usará como sistema operativo Windows o Linux y como navegadores Internet Explorer y Mozilla. La aplicación brindará una interfaz web por lo que necesitará una computadora con conexión a servidores externos. La conexión entre la PC Cliente y la PC Servidor Web es mediante el protocolo HTTP, así como la relación entre las PC servidoras es con el protocolo TCP/IP.

### 3.2.2. Diagrama de Componentes

Un diagrama de componentes muestra las organizaciones y dependencias lógicas entre componentes software, sean éstos componentes de código fuente, binarios o ejecutables. Normalmente contienen componentes, interfaces y relaciones entre ellos y como todos los diagramas, también puede contener paquetes utilizados para agrupar elementos del modelo.





### 3.3. Seguridad

La seguridad como elemento esencial para preservar la integridad, autenticidad y confiabilidad de la información debe estar presente en cada sistema de software que se desarrolle, por ello y debido a que el sistema implementado está basado en la web, la seguridad juega un papel fundamental para el correcto funcionamiento.

El uso de Django garantiza la protección automática de muchos de los errores comunes de seguridad, brindando facilidades y herramientas para combatir vulnerabilidades como inyecciones SQL, Cross-Site Scripting (XSS), inyección de cabeceras de email, exposición de mensajes de error, entre otras facilidades.

El sistema provee además un sistema de autenticación donde se solicita nombre de usuario y contraseña verificando que los datos introducidos sean válidos y garantizando que usuarios no autorizados accedan a los datos que se manejan en el sistema.



En este capítulo se ha concluido el proceso de implementación, donde se obtuvieron los artefactos Diagrama de despliegue y Diagrama de componentes que conforman el Modelo de implementación. Se obtuvo además la descripción de las tablas que forman parte del modelo de datos. Se implementó el proceso de Gestión de Investigaciones de acuerdo a los requerimientos de seguridad necesarios para garantizar la confidencialidad y privacidad de la información que se manipula en todo el sistema.



---

## CONCLUSIONES

Con el desarrollo de la presente investigación se analizaron los procesos del negocio asociados al área de gestión de las investigaciones y se determinó la necesidad de desarrollar un sistema informático para la gestión de la misma. El desarrollo de las tareas de la investigación permitió dar cumplimiento al objetivo general y arribar a las siguientes conclusiones:

- Se demostró la necesidad existente de desarrollar un sistema informático que permita viabilizar la gestión de la información que se genera en los procesos relacionados con la gestión investigativa en la Facultad 7 de la Universidad de las Ciencias Informáticas, ya que los sistemas estudiados no satisfacen, de forma individual, las necesidades que implican la gestión de la información.
- Se asimiló la arquitectura definida para el desarrollo del Sistema Integral de Gestión Administrativa.
- Se documentaron los flujos de trabajo: Modelamiento de Negocio, Requerimientos, Diseño e Implementación de los procesos relacionados con la gestión investigativa.
- Se realizó la implementación de una aplicación web que permita viabilizar el proceso de gestión de la información relacionada con la gestión investigativa en la Facultad 7 de la Universidad de las Ciencias Informáticas.



---

## RECOMENDACIONES

Luego de culminada la investigación y cumplido el objetivo general de la misma, en vista a mejorar el funcionamiento del sistema, se recomienda:

- Dar la posibilidad de gestionar la información de estudiantes, para así lograr un mayor control de los procesos investigativos de la Facultad 7 en la Universidad de las Ciencias Informáticas.
- Dar la posibilidad de exportar los reportes a otros formatos como documentos Word, Excel.
- Adicionar funcionalidades que permitan almacenar información referente a la superación profesional.
- Instalar el sistema en las demás facultades de la Universidad de las Ciencias Informáticas y otras universidades.
- Cambiar el diseño actual por uno que sea más atractivo y amigable para el usuario.
- Instalar el sistema en las demás facultades de la Universidad de las Ciencias Informáticas u otras universidades.



---

**REFERENCIAS**

1. **León Robaina, Rosario.** [Online] <http://www.udual.org/CIDU/Revista/22/RetosDesafios.htm>.
2. **Alvarez Zayas, Carlos.** *Metodología de la Investigación.* Ciudad de la Habana : ECIMED, 1989.
3. **Información Tecnológica.** [Online] [http://www.scielo.cl/scielo.php?pid=S0718-07642007000100014&script=sci\\_arttext](http://www.scielo.cl/scielo.php?pid=S0718-07642007000100014&script=sci_arttext).
4. **Cooperación Universitaria, Oficina.** [Online] [http://menweb.mineducacion.gov.co/Archivos\\_Comunica/html/doc/OCU.pdf](http://menweb.mineducacion.gov.co/Archivos_Comunica/html/doc/OCU.pdf).
5. **Universidad de Medellín.** [Online] <http://www.udem.edu.co/UDEM/Investigacion/Políticas/SIDI.htm>.
6. [Online] <http://www.um.es/docencia/barzana/IAGP/lagp2.html#BM1>.
7. **Fowler, Martín.** *UML Gota a Gota.* 1999.
8. **Ilver, Anache and Moreno, Joel.** [Online] <http://www.avatarsrl.com/website/articulos/AVATAR%20-%20Articulo%20OMG%20UML%202.pdf>.
9. **WingWare.** [Online] <http://wingide.com/wingide>.



---

**BIBLIOGRAFÍA**

1. **Alvarez Zayas, Carlos.** *Metodología de la Investigación.* Ciudad de la Habana : ECIMED, 1989.
2. **Comunidad Linperial de Desarrolladores.** [Online] <http://www.linperial.com/communities/forums/developers/?q=node/62>.
3. **Cooperación Universitaria, Oficina.** [Online] [http://menweb.mineduacion.gov.co/Archivos\\_Comunica/html/doc/OCU.pdf](http://menweb.mineduacion.gov.co/Archivos_Comunica/html/doc/OCU.pdf).
4. **Dirección de Investigación.** [Online] <http://di.atalca.cl/>.
5. **Dirección de Investigaciones Científica y Tecnológica.** [Online] <http://www.dicyt.umss.edu.bo/>.
6. **Fowler, Martín.** *UML Gota a Gota.* 1999.
7. **Ilver, Anache and Moreno, Joel.** [Online] <http://www.avatarsrl.com/website/articulos/AVATAR%20-%20Articulo%20OMG%20UML%202.pdf>.
8. **Información Tecnológica.** [Online] [http://www.scielo.cl/scielo.php?pid=S0718-07642007000100014&script=sci\\_arttext](http://www.scielo.cl/scielo.php?pid=S0718-07642007000100014&script=sci_arttext).
9. **León Robaina, Rosario.** [Online] <http://www.udual.org/CIDU/Revista/22/RetosDesafios.htm>.
10. [Online] <http://e-archivo.uc3m.es/dspace/bitstream/10016/4945/1/Tesis.pdf>.
11. [Online] [http://menweb.mineduacion.gov.co/Archivos\\_Comunica/html/doc/OCU.pdf](http://menweb.mineduacion.gov.co/Archivos_Comunica/html/doc/OCU.pdf).
12. [Online] [http://www.ocu.es/portal/pls/portal/!PORTAL.wwpob\\_page.show?\\_docname=322280.PDF](http://www.ocu.es/portal/pls/portal/!PORTAL.wwpob_page.show?_docname=322280.PDF).
13. [Online] <http://www.um.es/docencia/barzana/LAGP/lagp2.html#BM1>.
14. [Online] <http://www.webmastergranada.es/descargas/Charlas/Python+Django.pdf>.
15. **Portal del Investigador.** [Online] [http://portaldelinvestigador.urjc.es/www/Que\\_es\\_UXXI.html](http://portaldelinvestigador.urjc.es/www/Que_es_UXXI.html).
16. **Phyton.** [Online] <http://www.python.org/>.
17. **Sparx Systems.** [Online] <http://www.sparxsystems.com.ar/products/ea.html>.
18. **Sistema de Gestión de la Investigación.** [Online] <http://sgi.atalca.cl/sgi.htm>.
19. **Universidad de Medellín.** [Online] <http://www.udem.edu.co/UDEM/Investigacion/Políticas/SIDI.htm>.
20. **Web-Saiprhi.** [Online] [http://proyectos.dicyt.umss.edu.bo/investigacion\\_umss/inicio.php](http://proyectos.dicyt.umss.edu.bo/investigacion_umss/inicio.php).
21. **WingWare.** [Online] <http://wingide.com/wingide>.