

The logo for Universidad de las Ciencias Informáticas (UCi) consists of the letters 'UCi' in a bold, sans-serif font. The 'U' and 'C' are dark blue, while the 'i' is a lighter blue with a small square dot above it.

Universidad de las  
Ciencias Informáticas



*Autores: Dania Mora Valdés  
Yoinel Ramos LaO*

*Trabajo de diploma para optar por el Título de Ingeniero  
en Ciencias Informáticas*

***PLATAFORMA E INTRANET PARA EL CONTROL  
INTERNO DE LOS RECURSOS HUMANOS DE LA  
FACULTAD 9***

*Tutor(a): Ms.C. Yaneisis Pérez Heredia*

*Co-tutor(es): Ing. Dennis Meriño Menadier*

*Ing. Yoandrys S. Pacheco Jérez*

*Ciudad de la Habana, Julio, 4 del 2008*

*“Los buenos comienzos propician buenos finales...”*

*Dania*

*Dedico mi tesis en primer lugar a mis padres.*

*A mi tía Lucy.*

*A la memoria de mis abuelos y de mi tía Nenita.*

*A mis amigos, a mi novio.*

*A Fidel, a la figura del Ché, a la revolución cubana y a la UCI.*

*Yoinel*

*Dedico este trabajo de Diploma especialmente a mi princesita Lissdania.*

*A mi niño Dalian, a mis padres, a mi mujer, a mi hermana, a mis abuelos.*

*A mis tíos, a mis primos, y sobrinos.*

*A todas las personas que se han esforzado por ser de mí un gran profesional.*

*A las personas que han tenido para mí un buen consejo, una palabra cariñosa, y que me han brindado apoyo en lo largo de la carrera.*

*A mis compañeros y a mis amigos.*

*A Fidel a Raúl, a la Revolución y a la UCI.*

*“gracias por todo aquello que vi, cuando creías que no estaba mirando”*

*Dania*

*Agradezco a mis padres por la buena educación y enseñanzas que me inculcaron, a mis tías y abuelas que estuvieron conmigo desde niña guiándome en mi camino.*

*A mi novio por su apoyo y cariño.*

*A mis amigos por estar en los buenos y en los malos momentos, en especial a Marilia, Belayne, Jorge Daniel y Bernardo.*

*A Fidel por ser el faro de luz de nuestro pueblo y por tener la idea de crear una universidad como la nuestra.*

*A mi tutora y cotutores por toda la ayuda recibida.*

*Agradezco a todas las personas que en estos 5 años me han brindado su amistad y cariño, y a todo el que me ha ayudado de una u otra forma.*

*Yoinel*

*A mis padres por ser mi fuente de inspiración.*

*A mi adorada hija, por ser lo más grande que tengo en esta vida, por ser mi razón de ser.*

*A mi mujer por su apoyo, confianza, cariño, comprensión y dedicación.*

*A mi tutora, y cotutores por guiarme en todo momento en la realización del trabajo.*

*A la UCI por brindarme la oportunidad de superarme, por las cosas que pude aprender durante la carrera, gracias a sus buenos profesores, por los momentos y experiencias adquiridas, y las amistades surgidas.*

*A la Revolución Cubana y en especial a Fidel por tener esa gran visión del futuro, por ser nuestro guía y en todo momento incitarnos al continuo estudio y superación para ser mejores cada día.*

*Agradezco en sentido general a todas las personas que me han brindado su apoyo, y han aportado ideas y soluciones durante el desarrollo del trabajo.*

**DECLARACIÓN DE AUTORIA**

Declaramos que somos los únicos autores del presente trabajo. Autorizamos a la Universidad de las Ciencias Informáticas (UCI) para que hagan el uso que estimen pertinente con este trabajo.

Para que así conste firmamos la presente a los \_\_\_\_días del mes de \_\_\_\_\_ del 2008.

---

Dania Mora Valdés

---

Yoinel Ramos La O

---

Yaneisis Pérez Heredia

## RESUMEN

El objetivo de este trabajo es la creación de una plataforma para controlar los diferentes subsistemas de la Facultad 9, así como el diseño de una base de datos que la sustente. Se presentan los modelos de negocio, sistema, análisis, diseño, despliegue e implementación propuestos para dar una solución factible al problema existente utilizando como metodología de construcción de software, el Proceso Unificado de Desarrollo de Software (RUP). Se hace un estudio de los sistemas existentes, de las herramientas y de la arquitectura de software adecuada para la construcción del sistema.

Como resultado de la investigación se presenta un sistema que brinda la seguridad e integridad que requieren los datos a procesar y agiliza y facilita los procesos que se generan en las diferentes áreas de la facultad.

ÍNDICE

INTRODUCCIÓN.....	1
Capítulo 1. Fundamentación Teórica .....	6
INTRODUCCIÓN .....	6
1.1    CONCEPTOS ASOCIADOS AL SISTEMA DE GESTIÓN DE INFORMACIÓN DE LA FACULTAD .....	6
1.2    PLATAFORMA E INTRANET .....	12
1.3    DESCRIPCIÓN DEL SISTEMA DE GESTIÓN DE INFORMACIÓN DE LA FACULTAD.....	12
1.4    SITUACIÓN PROBLEMÁTICA .....	13
1.5    ANÁLISIS DE OTRAS PLATAFORMAS EXISTENTES .....	13
1.6    CONCLUSIONES PARCIALES.....	14
Capítulo 2. Tendencias actuales a desarrollar .....	16
INTRODUCCIÓN .....	16
2.1    DESARROLLO DE APLICACIONES WEB.....	16
2.2    GESTORES DE BASE DE DATOS .....	18
2.3    ¿POR QUÉ UTILIZAR POSTGRESQL CÓMO GESTOR DE BASE DE DATOS?.....	20
2.4    LENGUAJE DEL LADO DEL CLIENTE .....	22
2.5    LENGUAJES DE PROGRAMACIÓN AL LADO DEL SERVIDOR .....	22
2.5.1    ¿Por qué utilizar PHP cómo lenguaje de programación?.....	24
2.6    ECLIPSE PDT (PHP DEVELOPMENT TOOLS).....	25
2.7    METODOLOGÍAS DE DESARROLLO DE SOFTWARE.....	26
2.8    ¿POR QUÉ USAR EL PROCESO UNIFICADO DE DESARROLLO DE SOFTWARE (RUP) COMO BASE EN EL DESARROLLO DE LA SOLUCIÓN? .....	27
2.9    EL LENGUAJE UNIFICADO DE MODELADO (UML) COMO SOPORTE DE LA MODELACIÓN DE LA SOLUCIÓN PROPUESTA.....	29
2.10    HERRAMIENTAS PARA MODELADO DE SISTEMA .....	30
2.10.1    ¿Por qué usar Rational Rose Enterprise Edition cómo herramienta de modelado? 32	
2.11    HTTP APACHE COMO SERVIDOR A UTILIZAR .....	33
2.12    FRAMEWORKS DE PHP .....	33
2.12.1    ¿Por qué utilizar Symfony como framework PHP?.....	35
2.13    OTRAS HERRAMIENTAS UTILIZADAS .....	38
2.14    CONCLUSIONES PARCIALES.....	38
Capítulo 3. Modelo de Negocio y de Sistema .....	39
INTRODUCCIÓN .....	40
3.1    ¿QUE ES UN MODELO?.....	40
3.2    MODELO DEL NEGOCIO DEL SUBSISTEMA DE RRHH .....	41
3.2.1    Actores y Trabajadores del Negocio .....	41
3.2.2    Proceso de Negocio del Subsistema de RRHH .....	42
3.2.3    Diagramas de Casos de Uso del Subsistema de RRHH.....	42
3.2.4    Descripción Textual de los Casos de Uso del Negocio .....	43
3.3    MODELO DE OBJETOS DEL SUBSISTEMA DE RRHH .....	44
3.4    REGLAS DEL NEGOCIO .....	44
3.5    REQUERIMIENTOS FUNCIONALES DE LA PLATAFORMA.....	45
3.6    REQUERIMIENTOS NO FUNCIONALES DE LA PLATAFORMA .....	45
3.7    MODELO DE SISTEMA .....	47
3.7.1    Descripción de los actores de la plataforma.....	47
3.7.2    Casos de Uso de la plataforma.....	51
3.7.3    Descripción de los Casos de Uso de la plataforma.....	52
3.7.4    Diagrama de Caso de Uso de la Plataforma .....	67
3.8    CONCLUSIONES PARCIALES.....	69

Capítulo 4. Construcción del Sistema.....	71
INTRODUCCIÓN .....	71
4.1 ANÁLISIS DE LA INTRANET Y EL SUBSISTEMA RRHH.....	71
4.1.1 Diagrama de Clases del Análisis .....	71
4.2 DISEÑO DE LA INTRANET Y EL SUBSISTEMA RRHH.....	74
4.2.1 Diagramas de clases del diseño .....	75
4.3 PRINCIPIOS DE DISEÑO .....	83
4.3.1 Estándares de la interfaz de la intranet y el subsistema RRHH.....	83
4.3.2 Tratamiento de excepciones y seguridad del Sistema.....	83
4.3.3 Patrones para asignar responsabilidades .....	83
4.4 INTEGRACIÓN CON SERVICIOS WEB .....	85
4.5 DISEÑO DE LA BASE DE DATOS DE LA PLATAFORMA.....	86
4.6 MODELO DE DESPLIEGUE DEL SUBSISTEMA RRHH.....	87
4.7 MODELO DE IMPLEMENTACIÓN DEL SUBSISTEMA RRHH.....	88
4.8 CONCLUSIONES PARCIALES.....	96
CONCLUSIONES.....	97
RECOMENDACIONES .....	98
REFERENCIAS BIBLIOGRÁFICAS.....	99
GLOSARIO DE TÉRMINOS.....	104
ANEXOS.....	106

## INTRODUCCIÓN

La introducción de las Tecnologías de la Informática y las Comunicaciones (TICs) impacta en todos los ámbitos del desempeño social, económico, político y en la cotidianeidad en la que el hombre se desarrolla. Las TICs constituyen, en consecuencia, uno de los elementos críticos para cualquier entidad. Su flexibilidad funcional y operativa, su soporte a los requerimientos organizacionales y sus capacidades de evolución son, entre otros, factores clave de éxito para el posicionamiento de cualquier institución. Las TICs han producido un cambio espectacular y drástico en todas las organizaciones, y en este sentido, es Internet el elemento más revolucionario. El uso creciente de Internet como instrumento de comunicación y de distribución de la información ha propiciado que su tecnología se adopte a nivel empresarial mediante la utilización de diversas herramientas informáticas propicias para la colaboración. De esta forma, por la necesidad de gestionar grandes volúmenes de datos y de comunicarse e interactuar eficiente y efectivamente en el interior de las organizaciones, surgió la intranet.

La intranet es una derivación de Internet, fruto de la aplicación de sus tecnologías en el interior de la organización, que surgen como respuesta a la necesidad de mejorar la productividad, reducir los costos y mantener los sistemas de información existentes. Su facilidad de uso y acceso ha permitido que una gran cantidad de usuarios la acepten como medio para obtener información confiable, de calidad y de sencilla manipulación a nivel interno de la empresa.

El uso adecuado de la intranet constituye un reto para la sociedad, en específico, para las organizaciones laborales que necesitan de ellas para mantener su competitividad, a la par que precisan cada vez más de Recursos Humanos (RRHH) capaces de usar las mismas en forma creadora.

En la UCI, creada en el año 2002, la primera institución de su tipo surgida de la Batalla de Ideas, al igual que otras instituciones de Cuba, debido a la gran información que se procesa y la cantidad de personas que trabajan y estudian en ella, se utiliza una de las herramientas más valiosas que tiene una organización, la Intranet. Esta permite a los estudiantes y trabajadores interactuar con los servicios que esta posee, hace más eficiente su trabajo, pero no recoge todos los datos necesarios para el funcionamiento de una facultad, ya que se centra en la comunidad universitaria en general.

La facultad 9 requiere de los servicios de una Plataforma que permita interconectar todas las áreas y que todos sus flujos de información se integren en una misma base de datos, de esta forma se podrá lograr mayor control en los RRHH, actividades e informaciones que se generen en la facultad.

El presente trabajo está encaminado a la creación de una Plataforma e Intranet para facilitar el control de los diferentes subsistemas que brindarán servicios a la facultad 9.

Todos los procesos que se realizan en las diferentes áreas de la facultad se llevan a cabo de forma manual, los datos son recogidos y archivados en copia dura. Esto conlleva a que puedan ocurrir errores, inconsistencia y ambigüedades en los datos así como en algunos casos pérdida de información. Debido a que todas las áreas requieren automatización, el sistema a construir estará dividido en los siguientes subsistemas:

- Subsistema de Extensión Universitaria y Residencia.
- Subsistema Docente.
- Subsistema de Departamentos Docentes.
- Subsistema de Producción.
- Subsistema de nóminas, evaluación de desempeño y entrega de materiales.
- Subsistema de Gestión de RRHH.

En la facultad 9 no existe ninguna aplicación con conexión a una base de datos que permita registrar, modificar y generar reportes de profesores y trabajadores que trabajen en la misma. El proceso de registro se realiza en un documento Excel, por lo que a la hora de realizar consultas en busca de alguna información específica que se quiera conocer de los profesores el proceso resulta complejo.

Por lo anteriormente planteado la investigación estaría encaminada a resolver el siguiente **problema científico**: Inexistencia de una Plataforma e Intranet para el control de los diferentes subsistemas que brindan servicios a la facultad 9.

Teniendo como **objeto de estudio** la Plataforma e Intranet enmarcada en el **campo de acción** del Sistema de gestión de la facultad.

Partiendo de lo antes mencionado se puede tomar como **idea a defender** que si se crea una Plataforma e Intranet que permita automatizar el control y la sustentabilidad de los diferentes subsistemas es posible elevar la eficiencia, la rapidez y la actualización en la documentación, eliminando así la ambigüedad y réplica de la información.

Para la realización de la investigación se definieron un conjunto de **Objetivos Específicos** que a continuación se muestran:

- Documentar el flujo de información entre las estructuras de dirección de la facultad que trabajan y operan con los RRHH.
- Establecer las modificaciones necesarias al flujo de información logrando su optimización máxima.
- Diseñar la base de datos necesaria para sustentar el sistema a desarrollar.

- Diseñar e implementar una Plataforma e Intranet que de respuesta al flujo de información de control de los recursos humanos de la facultad.

Para dar cumplimiento a estos objetivos se trazaron las siguientes **tareas**:

- Estudio de las principales y más actuales tecnologías utilizadas para el diseño y desarrollo de Plataformas e Intranet's en Cuba y el mundo.
- Definición de los estándares nacionales e internacionales para el desarrollo de Plataformas e Intranet's.
- Evaluación de las condiciones actuales en las cuales se desarrolla el proceso de Gestión de los RRHH en la Facultad.
- Diseño del flujo de información, sus nodos centrales, así como los datos de entrada y reportes para la actividad de gestión de los RRHH y el flujo de la información que se maneja.
- Diseño de la base de datos.
- Confección de la documentación completa UML del sistema, como sustento para lograr un software con calidad.
- Implementación de un sistema funcional para Gestionar el Control Interno de los RRHH.

Para realizar las tareas mencionadas se utilizará la estrategia de investigación explicativa o experimental, cuyo objetivo es realizar un estudio de todos los subsistemas que formarán parte de la plataforma en la facultad 9.

Finalmente se muestran los métodos científicos que dan sustento a la investigación realizada.

### **Métodos teóricos:**

**Histórico-Lógico:** para la realización del presente trabajo se comenzó realizando un estudio detallado de las plataformas que existen en el mundo, en Cuba y en particular en la UCI.

**Hipotético-Deductivo:** este método se utiliza porque a partir de la investigación realizada sobre las herramientas a utilizar se dedujo cuáles son las más factibles para la realización de una plataforma que le de sustento a la intranet de la facultad y a los diferentes subsistemas.

### **Métodos empíricos**

Entrevistas realizadas a los clientes, de carácter individual.

La búsqueda en Internet y en documentos relacionados con el tema.

Con el estudio realizado se espera como resultado obtener una Plataforma e Intranet, que controle los diferentes subsistemas que le brindan información a la facultad, pero además que cumpla con los requisitos solicitados por los clientes y les facilite su uso y control resolviendo así el problema planteado. Además de que el sistema tenga un ambiente multiusuario y una base de datos centralizada con todo los datos referentes a los subsistemas.



# *CAPÍTULO 1*

## *FUNDAMENTACIÓN TEÓRICA*

*“Aquel que duda y no investiga, se torna no sólo  
infeliz, sino también injusto.”*

*Blas Pascal.*

## Capítulo 1. Fundamentación Teórica

### Introducción

En el presente capítulo se definen algunos conceptos básicos relacionados al dominio del problema, de igual manera se hará una descripción del objeto de estudio, la situación problemática y el análisis de otras soluciones existentes, que servirán de ayuda para realizar el sistema.

### 1.1 Conceptos asociados al Sistema de Gestión de Información de la Facultad

A continuación se describen algunos de los principales conceptos asociados al sistema:

#### RRHH

Los individuos se consideran un factor de producción. Son considerados como un recurso, y como todo recurso requieren ser gestionados.

Recurso: Es una fuerza que conviene optimizar [(1)]

Se denomina RRHH al trabajo que aporta el conjunto de los empleados de una organización. Pero lo más frecuente es llamar así a la función que se ocupa de seleccionar, contratar, formar, emplear y retener a los colaboradores de la organización.

Los RRHH son considerados la variable más importante para la generación de ventajas competitivas en una empresa; más importantes incluso que el capital y la tecnología. Por ello en cada empresa se aplica una política encaminada a mejorar la competitividad del personal.

#### Control

El control se considera la última etapa del proceso administrativo, aunque normalmente la planeación y el control están relacionados; incluso, algunos autores consideran que el control es parte de la planeación.

“Implica la medición de lo logrado en relación con el estándar y la corrección de las desviaciones, para asegurar la obtención de los objetivos de acuerdo con el plan”.

(HAROLD KOONTZ Y CYRIL O'DONNELL)

“Es la medición de los resultados actuales y pasados en relación con los esperados, ya sea total o parcialmente, con el fin de corregir, mejorar y formular nuevos planes”.

(MADDOCK)

### **Capital humano**

Es un término usado en ciertas teorías económicas del crecimiento para designar a un hipotético factor de producción dependiente no sólo de la cantidad, sino también de la calidad del grado de formación y productividad de las personas involucradas en un proceso productivo [(2)].

El Capital Humano es el acumulado de las capacidades productivas y competitivas, que una persona adquiere por acumulación de conocimientos generales o específicos.

### **Plataforma**

Desde el enfoque de la realización de sistemas informáticos, se le denomina plataforma a los diferentes ambientes creados para el desarrollo de software. Actualmente en el mundo existen diferentes tipos de plataformas de desarrollo para aplicaciones electrónicas y de escritorio. Hoy en día los ambientes de programación han ido dirigiéndose especialmente al bienestar del desarrollador y a la unificación en un mismo entorno de desarrollo, la creación de aplicaciones con sus prestaciones para diferentes campos de acción como lo son, la web y las aplicaciones de escritorio. [(3)].

En informática, una plataforma de desarrollo es el entorno de software común en el cual se desenvuelve la programación de un grupo definido de aplicaciones. Comúnmente se encuentra relacionada directamente a un sistema operativo; sin embargo, también es posible encontrarla ligada a una familia de lenguajes de programación o a una Interfaz de programación de aplicaciones. [(4)].

Intel define una plataforma como un conjunto integrado de componentes que hace posibles modelos de uso específicos. [(5)].

### **Tipos y características de las Plataformas**

Hay diferentes tipos de plataformas: éstas pueden ser sistemas muy sencillos hasta llegar a ser altamente complejos y articulados. Una vez comprobadas las ventajas del uso de una plataforma [(6)] es necesario evaluar las diferentes herramientas de que se dispone y analizar sus funcionalidades, sin dejar de tomar en cuenta las exigencias formativas.

En una primera fase de orientación, resulta de gran utilidad considerar las numerosas aportaciones teóricas de quienes se han ocupado del análisis y evaluación de las características principales de estos sistemas y que han resaltado los elementos fundamentales a la hora de seleccionar una plataforma [(7), (8), (9)].

Se tienen plataformas comerciales o propietarias, y por otro lado plataformas Open Source. Mientras que las primeras son realizadas por empresas de desarrollo de software para lograr su venta, las segundas son desarrolladas por una comunidad de informáticos que las pone a disposición sin costo alguno y cuya licencia permite el acceso al código fuente para mejorar el programa (licencia GNU GPL). Esta segunda opción resulta de particular interés para las organizaciones que se ocupan de la formación relacionada con la experimentación, como las universidades, y que necesitan investigar y experimentar a través de estas herramientas. De hecho, el software libre constituye un recurso verdaderamente valioso ya que es fácilmente accesible y disponible en comparación con plataformas comerciales que estas son extremadamente caras y de difícil acceso para los usuarios.

### **Intranet**

Una intranet es una red o un conjunto de redes informáticas interconectadas pertenecientes a una misma institución donde solo tendrán acceso los usuarios que pertenecen a la institución.

➤ **Visión Técnica:**

"intranet es un conjunto de sitios web que están instalados en la red interna de una institución o empresa y que permiten mostrar datos o documentos a cualquiera de los computadores conectados a ella".

**RYAN BERNARD, 1996**

➤ **Visión Funcional:**

Desde el punto de vista funcional, la intranet es un sistema para la gestión de información, desarrollado principalmente sobre las potencialidades del mundo World Wide Web, y que es de acceso limitado a la institución o empresa que lo crea o a quien ella autorice. [(3)]

➤ **Definición Organizacional:**

Una intranet es una organización de aprendizaje, que permite la integración de personas, procesos, procedimientos y principios para formar una cultura intelectualmente creativa que permita la implantación de la efectividad total de la organización. [(3)]

### **Beneficios de una Intranet**

El uso de este tipo de redes aporta múltiples beneficios:

- **Ahorro:** elimina documentos, formularios, manuales, tarifas, notas internas, bases de datos y demás documentación que obliguen al uso de papel y a la utilización de impresoras para la distribución de la información diaria de la empresa.

- Calidad: internet es la cantidad mientras que una intranet es la calidad. La red está cargada de información importante, pero el tiempo que se debe emplear para su localización es a menudo demasiado para ofrecer soluciones rápidas a las dificultades diarias. En la intranet el usuario encuentra todo lo que necesita ya que la información se encuentra mucho más organizada y seleccionada.
- Comunicación: el sistema de mensajería implementado puede ofrecer comunicaciones internas y externas, en tiempo real y diferido, y completamente integradas.
- Control: los datos importantes de la organización no estarán tan sólo en la mesa de un directivo o en la de un mando intermedio. Se sabrá en cada momento cómo se encuentra un proyecto ya que la información en una intranet se modifica y consulta en tiempo real.
- Colaboración: permiten aprovechar la experiencia intelectual individual de todos los empleados y tenerla disponible para su utilización global. Los usuarios pueden crear y manejar sus propios contenidos.
- Efectividad: permite que no se dejen asuntos pendientes y que se acaben las excusas de falta de información para llevar a cabo un proyecto cualquiera. Además cada departamento genera y mantiene su propia documentación así como los datos que le pertenecen.
- Eficiencia: una intranet elimina en porcentajes muy altos los métodos de trabajo tradicionales, y en ocasiones poco eficientes como el teléfono, el fax o las pequeñas reuniones de pasillo.
- Facilidad: con el simple requerimiento de saber manejar un navegador, cualquier usuario corporativo podrá trabajar con una intranet. Con algo más de conocimientos, no excesivos, podrá dotarla de toda su información y compartirla con el resto de los empleados de la organización.
- Flexibilidad: el trazado de puesta en marcha de una intranet es algo que puede comenzar con un nivel relativamente simple e incrementar las prestaciones según las necesidades y las posibilidades de la empresa. El adecuado mantenimiento y la actualización posterior juegan un papel fundamental en el buen desarrollo del proyecto que se verá culminado con la adaptación al nuevo medio de todos los usuarios que componen la compañía.
- Rendimiento: todo el material que se puede tener de manera impresa en un catálogo, manual o libro, se puede implementar sin mucho esfuerzo en una intranet. [(10)]

### ¿Por qué usar Intranet?

El uso de una intranet o más bien la creación de ella, implica pensar de nuevo las cosas que en la empresa se hacen y ver el beneficio de implantar un nuevo esquema

- Aglutinar el acceso a la información

Una intranet permite aglutinar la información que generan los distintos departamentos o secciones de la organización. Esto implica la posibilidad de consultar por un dato que puede estar localizado en distintas fuentes de información que la empresa posee.

- Identificar los flujos de información

El uso de una intranet ayuda a identificar quién genera y quién usa la información que maneja la empresa. Las empresas saben quien produce información útil para su gestión.

Esta, permite una visión concentrada y organizada de los flujos de información, acotando los requerimientos del usuario.

- Economizar recursos a la empresa

Las empresas que sustentan su gestión en sistemas de computación, están conscientes de la importancia y costo que implica el capacitar a su personal en el uso de las aplicaciones computacionales para la producción y control de sus productos y/o servicios. Esto adquiere vital relevancia en los procesos administrativos y de gestión en los que usualmente el personal utiliza diversos programas con diferentes comandos para el logro de sus funciones, permite economizar recursos de entrenamiento de personal cuando es capaz de ofrecer una interfaz estandarizada que incorpore y homologue la mayor parte de las aplicaciones computacionales requeridas para la gestión de la empresa. Esto reduce el número de personas que necesitan tener dominio en el manejo de diversos programas y aplicaciones. Del mismo modo, reduce el costo de adquisición de licencias para las aplicaciones afectadas por la intranet.

- Actualizar la información

La actualización de la información es otra de las razones por la que las intranets son adoptadas en las organizaciones. El uso de una interfaz que permita a los miembros de la empresa generar y consultar información en tiempo real, otorga credibilidad al proceso administrativo y de gestión de la institución. Esto también permite dar una imagen corporativa al cliente, por cuanto él percibe que los datos con los que el interlocutor de la empresa trabaja, son actuales y por tanto confiables.

- Mejorar la gestión de RRHH

Quien administra personal, sabe que el logro de sus objetivos es el resultado de los acuerdos que adquiere con sus subalternos. Esto es, el acordar con ellos el logro de ciertos objetivos en determinado

plazo de tiempo, lo que naturalmente dependerá de ciertas variables de las cuales son responsables ellos mismos y otras personas que también trabajan en la empresa o que tienen una relación con ella. Al existir una actualización rápida de los datos, el administrador puede mejorar la gestión de sus recursos humanos porque puede monitorear lo que el trabajador de la empresa está haciendo para el logro de los acuerdos definidos. También el trabajador puede administrar mejor su tiempo porque tiene una noción real del estado de avance de las variables que lo afectan y con ello la oportunidad de sacar el mejor provecho de ellas.

➤ Fortalecer la comunicación

El punto anterior provoca un fortalecimiento de la comunicación en la empresa porque la información se sistematiza y fluye. Especifica quién, cómo y cuándo corresponde, no dependiendo de subjetividades, sino de los privilegios que el usuario de la intranet tiene respecto de ella. Estos privilegios son fruto de fronteras previamente definidas por los jefes o gerentes de la empresa. [(11)] [(12)]

### **A quién beneficia una Intranet**

Beneficia a la empresa porque optimiza los RRHH y materiales al ahorrar tiempo y dinero en capacitación, soporte o apoyo técnico y adquisición periódica de software y hardware. También permite reorientar los procesos que generan información y productos, para que trabajen considerando los intereses de cada parte de la empresa y no sólo los propios.

Beneficia al cliente con información oportuna y pertinente, fruto de la actualización constante de los datos con que la empresa trabaja y la convergencia de dichos datos mediante una sola interfaz, aunque provengan de distintas fuentes.

Beneficia a las personas que trabajan en la empresa, porque les permite contar con la información oportuna y confiable que requieren para su trabajo y tienen certeza que la información que generan, contribuye con otros dentro de la organización. En definitiva, se siente partícipe y comprometido con el proceso [(12)].

### **Tipos de Intranet**

El parámetro utilizado para su clasificación desde una perspectiva tecnológica, es la capacidad que se le atribuye para ingresar, modificar y consultar datos dentro del sistema. Bajo este prisma, se pueden distinguir dos tipos de intranets: las pasivas y las activas.

Aquellas denominadas pasivas, son interactivas pero no dinámicas, esto quiere decir, que no permiten el acceso a los datos y sólo permiten desplegar información estática en la pantalla.

Por el contrario, las activas son interactivas y son dinámicas, ya que permiten acceso e interacción con los datos por parte del usuario o cliente.

Visto desde una perspectiva funcional el parámetro de clasificación que se considera es, la función a la que sirve la intranet. En este caso, se encuentran: aquellas que apoyan a la toma de decisiones, control y gestión y, aquellas que apoyan a la parte operativa y administrativa de la organización. [(10; 13)]

### **1.2 Plataforma e Intranet**

EL objeto de estudio de esta investigación es la Plataforma e Intranet. La realización de una plataforma permite tener agrupados todos los subsistemas desarrollados para cada área con un único punto de acceso, la intranet, donde la información estará actualizada de manera rápida al alcance de todos los estudiantes y trabajadores de la facultad, mejorando así la toma de decisiones. Además la información será la misma para todos los usuarios. Todas las áreas tendrán la posibilidad de publicar información de interés para los estudiantes y trabajadores de la facultad 9.

### **1.3 Descripción del sistema de gestión de información de la facultad**

La facultad 9 requiere de los servicios de una Plataforma e Intranet que permita interconectar todas las áreas y que sus flujos de información se integren en una misma base de datos, para lograr un mayor control en los recursos humanos, actividades e informaciones que se generen en la misma. Cada área esta compuesta por estudiantes, profesores, trabajadores y por uno o varios responsables que son los encargados de velar por su funcionamiento.

Las áreas a informatizar son:

- Departamentos Docentes.
- Vicedecanato Formación.
- Vicedecanato Producción.
- Vicedecanato Residencia y Extensión Universitaria.
- Nóminas, Evaluación de desempeño y entrega de materiales.
- RRHH.

La plataforma incluirá como elemento básico una interfaz gráfica común, con un único punto de acceso teniendo diferentes usuarios con distintos privilegios, de manera que en ella se integran los diferentes subsistemas teniendo una sola base de datos general, con todos los datos referentes a cada área.

### 1.4 Situación problemática

La inexistencia de una Plataforma e Intranet en la facultad 9, afecta el trabajo que se lleva a diario en la misma, ya que todos los procesos que se generan en las diferentes áreas no están automatizados, la mayoría se realizan de forma manual. A continuación se describen algunos de los procesos a automatizar:

- El control de los estudiantes, los recursos y el estado de la residencia, así como los demás procesos relacionados con Extensión Universitaria.
- Las afectaciones que puede tener un profesor en la ubicación de sus turnos en el horario docente.
- El control de las licencias de matriculas y las licencias especiales.
- El reporte y control diario de la información, de vital importancia, para llevar a cabo el óptimo funcionamiento del trabajo del Departamento Docente.
- El control de la estadística, recursos materiales y humanos, así como el seguimiento de los proyectos productivos en el Vicedecanato de Producción.
- El registro y gestión de los datos de profesores y trabajadores de la facultad.
- El proceso de nómina y aseo de los profesores y trabajadores de la facultad.

Cada área trabaja por separado por lo que los flujos de información no están relacionados en ningún sistema, esto provoca descontrol, pérdida de información y poca coordinación, así como repetición de datos entre los diferentes directivos que velan por el funcionamiento de la facultad.

### 1.5 Análisis de otras Plataformas existentes

Con el avance de la Informática y las Comunicaciones en el mundo ha surgido la necesidad de crear herramientas que brinden soluciones a las necesidades de las empresas y organismos, dentro de estas herramientas se encuentran las plataformas.

En el ámbito internacional existen muchas plataformas en las universidades del mundo, pero la mayoría de las que se investigaron no agrupan todos los procesos de las diferentes áreas de una facultad o universidad, sino que su campo de acción se limitaba a un área en específico.

En las universidades del país se han creado diversas plataformas que permiten resolver situaciones concretas como el aprendizaje, educación a distancia, entre otras. En la investigación que se realizó en búsqueda de una plataforma que fuera capaz de vincular las diferentes áreas de una facultad se encontró que no se ha hecho mucha incursión en esta problemática. En la UCI existen plataformas que brindan servicios a un área en específico fundamentalmente relacionada con el aprendizaje de los

estudiantes y profesores como “moodle” pero no existe ninguna que englobe todo el flujo de información que se genera en una facultad.

### **1.6 Conclusiones Parciales**

Con el desarrollo de este capítulo, se profundizó en algunos conceptos necesarios para realización del Trabajo de Diploma, que están presentes o asociados de alguna forma con el dominio del problema, se han analizado otras soluciones existentes, se ha identificado el objeto de estudio, la descripción de cómo ocurren los procesos de gestión de la información de la facultad 9 en la actualidad, lo que se hace necesario que se realice un sistema que de solución a la problemática planteada.



## *CAPÍTULO 2*

### *TENDENCIAS Y TECNOLOGÍAS ACTUALES A DESARROLLAR.*

*“Con la tecnología y la lectura, el mundo  
será de personas que piensan.”*

## Capítulo 2. Tendencias actuales a desarrollar

### Introducción

En el mundo de hoy las exigencias de calidad son cada vez más rigurosas, las tendencias hacia los productos cada vez más personalizados, paralelamente a ello, la vida del producto es cada vez más corta por lo que para hacer uso de las tendencias y tecnologías actuales se necesitan personas con capacidad de adaptación al cambio y entrega total a la evolución de las nuevas tecnologías.

En este capítulo se describen todas las herramientas con las que se desarrollará la aplicación. Primeramente se realiza un análisis sobre la web, debido a que es el entorno en que se va a desarrollar, su importancia y sus ventajas. Se abordan las características del modelo cliente/servidor como arquitectura base a utilizar, Hypertext Pre-processor (PHP) como lenguaje de programación y PostgreSQL como gestor de base datos.

La metodología a utilizar será RUP, por lo que se exponen sus características y sus principales ventajas. Se dedica un epígrafe a las características del lenguaje de modelado de objetos (UML). Además se fundamenta Eclipse como Entorno de Desarrollo Integrado (IDE) de PHP.

Como estructura de soporte para desarrollar y organizar el sistema se utilizó Symfony, como Framework de PHP, por lo que en el capítulo se abordan sus características y su vinculación con el Modelo Vista-Controlador, además se describen otras herramientas utilizadas en el desarrollo del sistema.

### 2.1 Desarrollo de Aplicaciones Web

Las aplicaciones web como parte de las TICs están cada vez más presentes en internet, brindando un alto grado de disponibilidad de la información, haciéndose cada vez más dinámicas y ajustables a las necesidades de los usuarios, no solo como simples contenedores de información sino como verdaderos elementos integradores de las potencialidades de las aplicaciones de escritorios, multimedia, hipermedia y los conceptos positivos de la web de las generaciones anteriores.

A pesar de los diversos significados que tiene este término para los diferentes desarrolladores. En la actualidad y debido al auge de este tipo de aplicaciones se reconocen diferentes tipos de categorías en dependencia del tipo de negocio que modelen. Es importante destacar que las aplicaciones web tienen una arquitectura Cliente/Servidor, esta no es más que una forma de dividir las responsabilidades de un Sistema de Información separando la interfaz de usuario (Nivel de presentación) de la gestión de la información (Nivel de gestión de datos).

La separación entre cliente y servidor es una separación de tipo lógico, donde el servidor no se ejecuta necesariamente sobre una sola máquina ni es necesariamente un sólo programa. La arquitectura cliente/servidor sustituye a la arquitectura monolítica en la que no hay distribución, tanto a nivel físico como a nivel lógico.

En esta arquitectura los clientes pueden realizar funciones como son el manejo de la interfaz de usuario, que no es más que la forma en que los usuarios pueden comunicarse con una computadora, y comprende todos los puntos de contacto entre el usuario y el equipo, la captura y validación de los datos de entrada o la generación de consultas e informes sobre las bases de datos.

Por otra parte los servidores pueden realizar funciones como son la de gestionar los periféricos compartidos, el control de accesos concurrentes a bases de datos compartidas o enlaces de comunicaciones con otras redes de área local.

La seguridad es uno de los aspectos que se deben de tener en cuenta al desarrollar una aplicación web. Con respecto a este tema cabe destacar que no existe ningún método de seguridad perfecto, incluso en los softwares más complejos y los que brindan mayor protección se han detectados algunos ataques de personas no autorizadas las cuales han utilizado herramientas para burlar la seguridad. Por lo que queda claro que si se desea mantener la integridad de los datos y la información que se brinda hay que poner todo el empeño en implementar una buena seguridad. Entre las amenazas a las que están expuestas las aplicaciones web se encuentran:

- Manipular datos: los datos se ven amenazados tanto si están en tránsito como si están almacenados. Ya que a través de la red se pueden interceptar los paquetes que por ella viajan y luego son descifrados por los atacantes.
- Denegación de servicios: la disponibilidad y la confiabilidad de las aplicaciones se ven directamente afectadas por los ataques de denegación de servicio, esto hace que algunas de las aplicaciones no sean disponibles incluso para los usuarios autorizados a utilizar esos servicios. Una de las formas para evitar la denegación de servicio es, regular el límite de ancho de banda y de los recursos para evitar que un sitio web sea sobrecargado y bloquee todo un servidor. [(14)]

### **Ventajas de las aplicaciones web**

Entre las ventajas más significativas de las aplicaciones web se encuentran su forma de instalación y distribución, ya que normalmente su instalación consiste en configurar los componentes del lado del servidor sin necesidad de configuración del lado del cliente. Con aplicaciones basadas en web, todos los usuarios utilizan la misma versión.

Además su principal protocolo de comunicación es http, el cual funciona normalmente desconectado, es decir, el cliente hace una petición al servidor, este la procesa y devuelve el resultado terminando la comunicación entre ellos. Hoy en día ninguna compañía o centro educacional se plantea prescindir de una web que los identifique, del mismo modo que no se priva de las comunicaciones telefónicas o de la electricidad. La web permite una comunicación directa con público de cualquier lugar del mundo, las veinticuatro horas del día.

### **Estándares web**

Los estándares web son un conjunto de recomendaciones dadas por el World Wide Web Consortium (W3C) y otras organizaciones internacionales acerca de cómo crear e interpretar documentos basados en la web. Son un conjunto de tecnologías orientadas a brindar beneficios a la mayor cantidad de usuarios, asegurando la vigencia de todo documento publicado en la web. El objetivo es crear una web que trabaje mejor para todos, con sitios accesibles a más personas y que funcionen en cualquier dispositivo de acceso a Internet.

Un sitio basado en estándares web es compatible con todos los navegadores actuales, y lo será con versiones futuras, es más fácil de mantener y actualizar, el código es más simple, de esta forma se elimina la dependencia de un solo desarrollador, que permitirá mostrar una mayor consistencia visual. Gracias al uso de Lenguaje Extensible de Marcado de Hipertexto (XHTML) para el contenido y Cascading Style Sheets (CSS) para la apariencia, se puede transformar rápidamente un sitio, sin importar que se trate de una página Web o miles, realizando cambios en un solo lugar. [(15)]

Los documentos que separan apariencia de contenido usan menos código, además, CSS permite conseguir efectos que antes requerían el uso de Javascript e imágenes, por lo que los sitios basados en estándares utilizan menos ancho de banda y se muestran más rápido a los usuarios, mejorando dramáticamente la experiencia de estos.

Los documentos basados en XHTML válido son más relevantes para los motores de búsqueda, contienen mayor información y menos código, por lo que un sitio basado en estándares Web tendrá una mejor posición.

### **2.2 Gestores de base de datos**

El software de bases de datos ha experimentado un auge extraordinario a raíz de la creciente informatización de casi la totalidad de las empresas de hoy en día, rapidez, efectividad en los procesos y los grandes flujos de información son la necesidad más apremiante a la hora de optimizar servicios y productos. Ante esta notable demanda de soluciones informáticas han surgido multitud de gestores de

bases de datos, estos son programas que permiten manejar la información de modo sencillo y que prestan servicios para el desarrollo y el manejo de bases de datos.

**MySQL** es el gestor de bases de datos, probablemente, más usado en el mundo del software libre, debido a su gran rapidez y facilidad de uso.

Esta gran aceptación es debida, en parte, a que existen infinidad de librerías y otras herramientas que permiten su uso a través de gran cantidad de lenguajes de programación, además de su fácil instalación y configuración. [(16)]

Ventajas:

- Aprovecha la potencia de sistemas multiprocesador, gracias a su implementación multihilo.
- Soporta gran cantidad de tipos de datos para las columnas.
- Dispone de API's en gran cantidad de lenguajes (C, C++, Java, PHP, entre otros).
- Gran portabilidad entre sistemas.
- Soporta hasta 32 índices por tabla.
- Gestión de usuarios y contraseñas, manteniendo un muy buen nivel de seguridad en los datos.

Desventajas:

MySQL se incluye en el grupo de sistemas de bases de datos relacionales, pero carece de algunas de sus principales características:

- Subconsultas: tal vez ésta sea una de las características que más se echan en falta, aunque gran parte de las veces que se necesitan, es posible reescribirlas de manera que no sean necesarias.
- Select into table: esta característica propia de Oracle, todavía no está implementada.
- Triggers y Procedures: se tiene pensado incluir el uso de procedimientos almacenados en la base de datos, pero no el de triggers, ya que los triggers reducen de forma significativa el rendimiento de la base de datos, incluso en aquellas consultas que no los activan.
- Transacciones: a partir de las últimas versiones ya hay soporte para transacciones, aunque no por defecto (se ha de activar un modo especial).
- Integridad referencial: aunque sí que admite la declaración de claves ajenas en la creación tablas, internamente no las trata de forma diferente al resto de campos.

**PostgreSQL** es un sistema gestor de bases de datos relacionales orientadas a objetos con cerca de una década de desarrollo, soporta casi toda la sintaxis SQL (incluyendo subconsultas, transacciones, y tipos y funciones definidas por el usuario), contando también con un amplio conjunto de enlaces con lenguajes de programación (incluyendo C, C++, Java, Perl, Python, entre otros).

Entre sus principales características están:

- Control de Concurrencia Multi-Versión (MVCC )

MVCC, es la tecnología que PostgreSQL usa para evitar bloqueos innecesarios. Si alguna vez ha usado algún DBMS con capacidades SQL, tal como MySQL o Access, probablemente habrá notado que hay ocasiones en las que una lectura tiene que esperar para acceder a información de la base de datos. La espera está provocada por usuarios que están escribiendo en la base de datos. Resumiendo, el lector está bloqueado por los escritores que están actualizando registros.

Mediante el uso de MVCC, PostgreSQL evita este problema por completo. MVCC está considerado mejor que el bloqueo a nivel de fila porque un lector nunca es bloqueado por un escritor. En su lugar, PostgreSQL mantiene una ruta a todas las transacciones realizadas por los usuarios de la base de datos. PostgreSQL es capaz entonces de manejar los registros sin necesidad de que los usuarios tengan que esperar a que los registros estén disponibles.

- Cliente/Servidor

PostgreSQL usa una arquitectura proceso-por-usuario cliente/servidor. Hay un proceso maestro que se ramifica para proporcionar conexiones adicionales para cada cliente que intente conectar.

**Microsoft SQL Server** es un sistema de gestión de bases de datos relacionales (SGBD) basada en el lenguaje SQL, capaz de poner a disposición de muchos usuarios grandes cantidades de datos de manera simultánea.

Entre sus características figuran [(17)]:

- Cuenta con una gran variedad de herramientas administrativas y de desarrollo que permite mejorar la capacidad de instalar, distribuir, administrar y utilizar SQL Server.
- Puede utilizarse el mismo motor de base de datos a través de plataformas que van desde equipos portátiles que ejecutan Microsoft Windows 95 ò 98 hasta grandes servidores con varios procesadores que ejecutan Microsoft Windows NT, Enterprise Edition.
- Almacenamiento de datos.
- Incluye herramientas para extraer y analizar datos resumidos para el proceso analítico en línea (OLAP, Online Analytical Processing).
- Incluye también herramientas para diseñar gráficamente las bases de datos y analizar los datos mediante preguntas en lenguaje normal.

### 2.3 ¿Por qué utilizar PostgreSQL cómo gestor de base de datos?

Entre las razones por las que se selecciona PostgreSQL como sistema gestor de base de datos se pueden mencionar:

- Posee una gran escalabilidad
- Es capaz de ajustarse al número de CPUs y a la cantidad de memoria que posee el sistema de forma óptima, haciéndole capaz de soportar una mayor cantidad de peticiones simultáneas de manera diferente, más del doble que MySQL.
- Incorpora una estructura de datos array, además incluye herencia entre tablas por lo que se le incluye entre los gestores objeto-relacionales.
- Implementa el uso de rollback's, subconsultas y transacciones, haciendo su funcionamiento mucho más eficaz, y ofreciendo soluciones en campos en las que MySQL no podría.
- Se distribuye bajo licencia Berkeley Software Distribution (BSD), lo cual permite el uso y distribución sin costo, tanto para aplicaciones comerciales como no comerciales.
- Tiene la capacidad de almacenar procedimientos en la propia base de datos, equiparándolo con los gestores de bases de datos de alto nivel, como puede ser Oracle.
- Su resistencia a fallos, supera la prueba sin pérdida de datos.
- Es el segundo motor de base de datos con más funcionalidades tras Oracle.
- Esta considerado como la base de datos de código abierto más avanzada del mundo, pues proporciona un gran número de características que normalmente sólo se encontraban en las bases de datos comerciales tales como Oracle.[ (18)].

A continuación describe sus principales características:

- DBMS Objeto-Relacional

PostgreSQL aproxima los datos a un modelo objeto-relacional, y es capaz de manejar complejas rutinas y reglas. Ejemplos de su avanzada funcionalidad son consultas SQL declarativas, control de concurrencia multi-versión, soporte multi-usuario, transacciones, optimización de consultas, herencia, y arreglos.

- Altamente Extensible

PostgreSQL soporta operadores, funciones, métodos de acceso y tipos de datos definidos por el usuario.

- Soporte SQL Comprensivo.

PostgreSQL soporta la especificación SQL99 e incluye características avanzadas tales como las uniones (joins) SQL92.

- Integridad Referencial

PostgreSQL soporta integridad referencial, la cual es utilizada para garantizar la validez de los datos.

- Lenguajes Procedurales

PostgreSQL tiene soporte para lenguajes procedurales internos, incluyendo un lenguaje nativo denominado Procedural Language/PostgreSQL Structured Query Language (**PL/pgSQL**) el cual es un lenguaje imperativo provisto por el gestor de base de datos PostgreSQL.

Permite ejecutar comandos SQL mediante un lenguaje de sentencias imperativas y uso de funciones, dando mucho más control automático que las sentencias SQL básicas.

- Write Ahead Logging (WAL)

La característica de PostgreSQL conocida como Write Ahead Logging incrementa la dependencia de la base de datos al registro de cambios antes de que estos sean escritos en la base de datos. Esto garantiza que en el hipotético caso de que la base de datos se caiga, existirá un registro de las transacciones a partir del cual se podrá restaurar la base de datos.

Además PostgreSQL ofrece muchas ventajas para la realización de bases de datos respecto a otros sistemas, dentro de ellas se encuentran:

- Instalación ilimitada

Es frecuente que las bases de datos comerciales sean instaladas en más servidores de lo que permite la licencia. Con PostgreSQL, nadie puede demandarlo por violar acuerdos de licencia, puesto que no hay costo asociado a la licencia del software.

- Multiplataforma

PostgreSQL está disponible en casi cualquier Unix (34 plataformas en la última versión estable), y una versión nativa de Windows está actualmente en estado beta de pruebas. [(19)].

### 2.4 Lenguaje del lado del cliente

#### JavaScript

El JavaScript es un lenguaje de programación ampliamente utilizado en el mundo del desarrollo web por ser muy versátil y potente, tanto para la realización de pequeñas tareas como para la gestión de complejas aplicaciones es un lenguaje interpretado que, al contrario que las aplicaciones normales, que son ejecutadas por el sistema operativo, es ejecutado por el navegador para ver las páginas. Eso hace que se puedan desarrollar aplicaciones de muy diverso tipo, desde generadores de HTML, comprobadores de formularios, hasta programas que gestionen las capas de una página.

### 2.5 Lenguajes de programación al lado del servidor

El Lenguaje **PHP** es un lenguaje interpretado usado para la creación de aplicaciones para servidores, o creación de otro tipo de programas incluyendo aplicaciones con interfaz grafica usando la biblioteca GTK+.

Es un lenguaje del lado del servidor que permite la conexión a diferentes tipos de servidores de bases de datos tales como MySQL, PostgreSQL, Oracle, ODBC, DB2, Microsoft SQL Server, Firebird y SQLite; lo cual permite la creación de Aplicaciones web muy robustas

**ASP.NET** es un conjunto de tecnologías de desarrollo de aplicaciones web comercializado por Microsoft. Es usado por programadores para construir sitios web domésticos, aplicaciones web y servicios XML. Forma parte de la plataforma .NET de Microsoft y es la tecnología sucesora de la tecnología Active Server Pages (ASP).

Dentro de las principales características de ASP.NET se encuentran [(20)]:

- Eficiencia: desde el principio, uno de los objetivos más importantes del diseño de .NET ha sido su gran rendimiento y nivelación.
- Contenido y Código, por separado: separara la interfaz de usuario con el código.
- Soporte de Lenguajes: la posibilidad de escribir código en diferentes lenguajes es un alivio para los desarrolladores que en numerosas ocasiones, veían acotadas sus aplicaciones web, al estar obligados a trabajar con VBScript o JScript. ASP.NET soporta la programación en lenguajes potentes cómo, VisualBasic.Net (VB) y C#, el nuevo lenguaje creado por Microsoft con la intención de aprovechar la potencia del C++ y combinarlo con las facilidades que brinda a la programación en Internet un lenguaje como Java.
- Código Compilado: ya no interpreta el código como la hace la versión anterior de ASP. Dentro del entorno NGWS (New Generation Windows Services) el código es compilado **just-in-time**, logrando un enorme aumento en el rendimiento, a través de soporte nativo y servicios de caché
- Los nuevos Controles web Forms: adopta el modo de Visual Basic a la hora de utilizar controles. Esto permite separar el código de la interfaz del usuario de forma sencilla y clara.

**Java Server Pages (JSP)**, es un acrónimo de Java Server Pages, que en castellano vendría a decir algo como Páginas de Servidor Java. Es, pues, una tecnología orientada a crear páginas web con programación en Java.

Dicho lenguaje posibilita crear aplicaciones que se ejecuten en variados servidores web, de múltiples plataformas, ya que Java es en esencia un lenguaje multiplataforma, es un desarrollo de la compañía Sun Microsystems, y su funcionamiento se basa en scripts.

Unas de las principales características que tiene este lenguaje son [(21)]:

- JVM disponibilidad de código fuente: cuando hay problemas siempre se puede buscar el código fuente y saber qué es exactamente lo que está pasando detrás de las escenas.

- Seguridad: Java fue diseñado con la seguridad en mente. Proporciona sólidas bibliotecas para todas sus necesidades de seguridad.
- La carga dinámica de clases: capacidad de carga de clases de datos a través de diferentes dispositivos y la red.
- Generics: Java simplifica la codificación de los genéricos sin dejar de proporcionar el tipo de seguridad.

### Características

JavaScript comparte muchos elementos con otros lenguajes de alto nivel. Hay que tener en cuenta que este lenguaje es muy semejante a otros como C, Java o PHP, tanto en su formato como en su sintaxis, aunque por supuesto tiene sus propias características definitorias. Es un lenguaje que diferencia entre mayúsculas y minúsculas, por lo que si se escribe alguna expresión en minúsculas, se debe mantener esa expresión en minúsculas a lo largo de todo el programa se puede encerrar las expresiones que se escriban con una serie de caracteres especiales. Estos caracteres se denominan operadores y sirven tanto para encerrar expresiones como para realizar trabajos con ellas, como operaciones matemáticas o de texto.

Como JavaScript es un lenguaje de formato libre, se pueden escribir las líneas de código de la forma que considere mejor, aunque por supuesto se debe escribir siempre de la forma correcta.

Otro aspecto importante de JavaScript es la necesidad o no de utilizar el punto y coma ';' al final de las instrucciones. Este operador solo sirve para delimitar las instrucciones, pero aunque su uso es obligatorio en la mayoría de los lenguajes, es opcional en JavaScript. Si se usa este operador, se podrán incluir varias sentencias en la misma línea de código, y si no se usa, sólo se podría escribir una sentencia cada vez.

### 2.5.1 ¿Por qué utilizar PHP cómo lenguaje de programación?

Se decidió utilizar PHP como script del lado del servidor debido a que cuenta con una serie de ventajas respecto a otros como ASP, JSP, Perl por mencionar algunos, entre las ventajas se pueden citar:

Su alto rendimiento, bibliotecas incorporadas para muchas tareas web habituales, bajo costo, de fácil aprendizaje y uso. A continuación se explicarán algunos detalles de estas características.

- PHP corre en casi todas las plataforma utilizando el mismo código fuente, pudiendo ser compilado y ejecutado en muchas plataformas, incluyendo diferentes versiones de Unix, Windows (95, 98, NT, 2000, XP) y Macs.
- La sintaxis de PHP es similar a la del C, por esto cualquiera con experiencia en lenguajes del estilo C podrá entender rápidamente PHP.

- Está compuesto de un sistema principal, y un conjunto de módulos y una variedad de extensiones de código.
  - Presenta interfaces distintas para cada tipo de servidor. Actualmente se puede ejecutar bajo Apache, Internet Information Server (IIS), entre otros.
  - Puede ser configurado como módulo CGI (protocolo que puede ser usado para comunicarse entre formas web y programas).
  - PHP incorpora una gran cantidad de funciones integradas para realizar útiles tareas relacionadas con la web. Puede generar imágenes GIF al instante, establecer conexiones a otros servicios de red, enviar correos electrónicos, trabajar con cookies y generar documentos PDF, todo con unas pocas líneas de código. Generalmente es utilizado como módulo de Apache, lo que lo hace extremadamente veloz.
  - Está completamente escrito en C, así que se ejecuta rápidamente utilizando poca memoria.
- [(22)]

Otra de las características que hacen que PHP sea uno de los lenguajes más utilizados hoy en día en la web es la gran cantidad de documentación que posee y además que es Open Source, lo cual significa que el usuario no depende de una compañía específica para arreglar cosas que no funcionan, además no están forzados a pagar actualizaciones anuales para tener una versión que funcione.

### 2.6 Eclipse PDT (PHP Development Tools)

Un IDE es un editor de código que además puede servir para depurar y facilitar las diferentes tareas necesarias en el desarrollo de cualquier tipo de aplicación.

El IDE que se utilizará para la realización del presente trabajo es Eclipse PDT debido a que es una plataforma de software de código abierto independiente y que puede ser utilizado para varios lenguajes: Java, C, C++, PHP entre otros, además que tiene un completo sistema de ayuda que te muestra cómo son las funciones. [(23)]

Entre las características de Eclipse se encuentran:

- Editor sensible al contexto, el cual provee de resaltamiento de código, asistente de código y auto completado de código.
- Integración con el modelo del proyecto Eclipse, que permite inspeccionar el uso de las vistas del contorno del fichero y del proyecto, así como la nueva vista PHP Explorer.
- Compilación en tiempo real
- Fácil de usar intuitivo e integrable con la *Web Tools* de Eclipse, extensibilidad y soporte continuo de desarrolladores PHP.

- Soporte de las versiones 4 y 5 de PHP indistintamente, ya sea bien de forma genérica a todos los proyectos que se generen o bien de forma individual a cada uno con previa especificación en las propiedades del proyecto en cuestión.
- Soporte completo del sistema de documentación PHPDoc, como característica clave la ayuda contextual a la hora de editar la documentación.
- Gestión y exploración de todas las clases generadas a lo largo de la edición del código o bien que se hayan importado de otra librería de PHP, tales clases las toma Eclipse y pueden ser usadas en todo el proyecto como si fueran parte de la librería estándar de PHP.
- Informe de todos los fallos de sintaxis cometidos mientras se edita el código, aunque se puede modificar este comportamiento para que sea un poco más o menos estricto.
- Formatea el código para mayor legibilidad.
- Soporte para el debug incremental del código de PHP.
- Extensos frameworks y APIs que permiten a los desarrolladores e ISVs (vendedores de software independientes) fácilmente extender IDE para crear nuevas e interesantes herramientas orientadas al desarrollo de PHP. [(24)]

### 2.7 Metodologías de desarrollo de software

La disciplina de Ingeniería de Software ha evolucionado en gran medida, trayendo consigo propuestas diferentes para mejorar los resultados del proceso de construcción de soluciones informáticas. La realización de diagramas de modelado y de prototipos tienen el objetivo de mostrar y graficar las actividades a implementar según los requerimientos capturados, así como especificar el camino a seguir y exponer el resultado que se obtendrá de la implementación de cada uno de estos requerimientos del cliente una vez terminado el desarrollo, brindando además la posibilidad de obtener retroalimentación de manera temprana.

Para el desarrollo de soluciones informáticas existen diferentes metodologías, pero siempre escoger una u otra se vuelve difícil.

#### **XP (Extreme Programming).**

Es una de las metodologías de desarrollo de software más exitosas en la actualidad, utilizadas para proyectos de corto plazo, equipo pequeño y cuyo plazo de entrega era ayer (entrega inmediata). La metodología consiste en una programación rápida o extrema, cuya particularidad es tener como parte del equipo, al usuario final, pues es uno de los requisitos para llegar al éxito del proyecto [(25)]

Características de XP, la metodología se basa en:

- Pruebas Unitarias: se basa en las pruebas realizadas a los principales procesos, de tal manera que adelantándonos en algo hacia el futuro, se puedan realizar pruebas de las fallas que pudieran ocurrir. Es como adelantarse a obtener los posibles errores.
- Refabricación: se basa en la reutilización de código, para lo cual se crean patrones o modelos estándares, siendo más flexible al cambio.
- Programación en pares: una particularidad de esta metodología es que propone la programación en pares, la cual consiste en que dos desarrolladores participen en un proyecto en una misma estación de trabajo. Cada miembro lleva a cabo la acción que el otro no está haciendo en ese momento.

### **MSF (Microsoft Solution Framework).**

Es una metodología flexible e interrelacionada con una serie de conceptos, modelos y prácticas de uso, que controlan la planificación, el desarrollo y la gestión de proyectos tecnológicos. MSF se centra en los modelos de proceso y de equipo dejando en un segundo plano las elecciones tecnológicas.

MSF tiene las siguientes características [(25)]:

- Adaptable: es parecido a un compás, usado en cualquier parte como un mapa, del cual su uso es limitado a un específico lugar.
- Escalable: puede organizar equipos tan pequeños entre tres o cuatro personas, así como también, proyectos que requieren cincuenta personas o más.
- Flexible: es utilizada en el ambiente de desarrollo de cualquier cliente.
- Tecnología Agnóstica: porque puede ser usada para desarrollar soluciones basadas sobre cualquier tecnología.

### **2.8 ¿Por qué usar El Proceso Unificado de Desarrollo de Software (RUP) como base en el desarrollo de la solución?**

A partir del estudio realizado a las principales metodologías para el desarrollo de la aplicación, se decidió que la metodología idónea a seguir es RUP, ya que es una de las más aplicadas en la actualidad para los proyectos grandes y complejos. A continuación se describe el porque utilizar esta metodología:

RUP es un proceso iterativo e incremental de Ingeniería de Software el cual designa tareas y responsabilidades. Asegura la producción de software de alta calidad y es capaz de ajustarse a las necesidades de los usuarios finales con un costo y un calendario predecible.

RUP tiene tres características esenciales:

- Está dirigido por los Casos de Uso: son una herramienta para especificar los requisitos del sistema. También guían su diseño, implementación y prueba. Los Casos de Uso constituyen un elemento integrador y una guía del trabajo.
- Está centrado en la arquitectura: toma en consideración elementos de calidad del sistema, rendimiento, reutilización y capacidad de evolución por lo que debe ser flexible durante todo el proceso de desarrollo. La arquitectura se ve influenciada por la plataforma software, sistema operativo, gestor de bases de datos, protocolos, consideraciones de desarrollo como sistemas heredados.
- Es iterativo e incremental: consta de una secuencia de iteraciones, cada iteración aborda una parte de la funcionalidad total, pasando por todos los flujos de trabajo relevantes y refinando la arquitectura. Cada iteración se analiza cuando termina. Se puede determinar si han surgido nuevos requisitos o han cambiado los existentes, afectando a las iteraciones siguientes.

Vale mencionar que el ciclo de vida que se desarrolla por cada iteración, es llevada bajo dos disciplinas

- Disciplina de Desarrollo.
- Disciplina de Soporte.

Los elementos del RUP son:

- Trabajadores (“Quién”): define el comportamiento y responsabilidades (rol) de un individuo, grupo de individuos, sistema automatizado o máquina, que trabajan en conjunto como un equipo. Ellos realizan las actividades y son propietarios de elementos.
- Actividades (“Cómo”): es una tarea que tiene un propósito claro, es realizada por un trabajador y manipula elementos.
- Artefactos (“Qué”): productos tangibles del proyecto que son producidos, modificados y usados por actividades. Pueden ser modelos, elementos dentro del modelo, código fuente y ejecutables.
- Flujo de Actividades (“Cuándo”): secuencia de actividades realizadas por trabajadores y que produce un resultado de valor observable.

Una particularidad de esta metodología es que, en cada ciclo de iteración, se requiere el uso de artefactos, siendo por este motivo, una de las metodologías más importantes para alcanzar un grado de certificación en el desarrollo del software [(26)].

### **2.9 El Lenguaje Unificado de Modelado (UML) como soporte de la modelación de la solución propuesta**

El aumento de la complejidad de los sistemas informáticos representa un desafío importante para los ingenieros y arquitectos del software. De la preocupación inicial sobre la definición de la estructura y calidad del código final, se ha pasado a dedicar cada vez más tiempo, atención y esfuerzo al análisis, diseño y modelado de los sistemas.

Los modelos proporcionan un mayor nivel de abstracción, permiten trabajar con sistemas mayores y más complejos, facilitando el proceso de codificación e implementación del sistema de forma distribuida y en distintas plataformas.

Entre los lenguajes de modelado que define Object Management Group (**OMG**) el más conocido y usado es sin duda Unified Modelling Language (**UML**).

UML es un lenguaje gráfico para especificar, construir y documentar los artefactos que modelan un sistema, ofrece un estándar para describir un modelo del sistema, incluyendo aspectos tales como procesos de negocios y funciones del sistema, así como aspectos concretos, dígame expresiones de lenguajes de programación, esquemas de bases de datos y componentes de software reutilizables.

UML fue diseñado para ser un lenguaje de modelado de propósito general, por lo que puede utilizarse para especificar la mayoría de los sistemas basados en objetos o en componentes, para modelar aplicaciones de muy diversos dominios de aplicación (telecomunicaciones, comercio, sanidad, entre otras.), plataformas de objetos distribuidos (como por ejemplo J2EE, .NET o CORBA) y además se

puede aplicar en una gran variedad de formas para soportar una metodología de desarrollo de software como por ejemplo el Proceso Unificado de Rational, pero no especifica en sí mismo, qué metodología o proceso utilizar.

Es un estándar que establece una plataforma de desarrollo de sistemas distribuidos, facilitando la invocación de métodos remotos bajo un paradigma orientado a objetos [ (27)]

En UML existen diferentes tipos de diagramas, los que se organizan jerárquicamente para una mejor comprensión. Ellos son:

- Diagramas de estructura: enfatizan en los elementos que deben de existir en el modelado.
- Diagramas de comportamiento: enfatizan en lo que debe suceder en el sistema modelado.
- Diagramas de Interacción: un subtipo de diagramas de comportamiento, que enfatiza sobre el flujo de control y de datos entre los elementos del sistema modelado.

### **2.10 Herramientas para Modelado de Sistema**

Para el modelado de UML es preciso el empleo de Herramientas Computer Aided Software Engineering (**CASE**). Estas herramientas ayudan en todos los aspectos del ciclo de vida de desarrollo del software en tareas como el proceso de realizar captura de requisitos, análisis y modelado, diseño del proyecto, cálculo de costos, implementación, compilación automática, documentación o detección de errores, entre otras.

Tienen como sus principales objetivos:

- Mejorar la productividad en el desarrollo y mantenimiento del software.
- Aumentar la calidad del software.
- Mejorar el tiempo y costo de desarrollo y mantenimiento de los sistemas informáticos.
- Mejorar la planificación de un proyecto.
- Aumentar la biblioteca de conocimientos informáticos de una empresa ayudando a la búsqueda de soluciones para los requisitos.
- Automatizar el desarrollo del software, la generación de código y gestión del proyecto.
- Ayudar a la reutilización del software y a la portabilidad y estandarización de la documentación.
- Gestión global en todas las fases de desarrollo de software con una misma herramienta.
- Facilitar el uso de las distintas metodologías propias de la ingeniería del software. [(28)]

A continuación se describen algunas de las principales características de las herramientas CASE más utilizadas en la actualidad:

### **Visual Paradigm para UML**

- Permite realizar los principales diagramas UML, aunque restringe a un solo tipo de diagrama en cada modelo.
- Validación en tiempo real del modelo.
- Interfaz de usuario configurable.
- Estructuración u organización automática de diagramas.
- Copia de diagramas como imágenes.
- Soporta subdiagramas para todos los modelos UML.
- Provee pleno soporte para ingeniería directa e inversa.

### **Poseidón**

- Poseidón Community Edition, esta versión permite generar los diagramas básicos de UML, aunque en un ambiente no profesional, tiene restringidas opciones como copia y pega al Clipboard de Windows, de modo que no permite copiar los diagramas como imágenes a presentaciones o documentos. Permite la generación de código en Java pero no soporta la ingeniería inversa.
- Poseidón Edición Estándar, permite entre sus funcionalidades principales la generación de documentación automática, ingeniería directa e inversa para código Java, soporta plataformas como Windows, Linux y Mac.
- Además posee un mecanismo de plugins que permite una configuración alta en función de las necesidades del usuario.
- Poseidón Edición Profesional, permite ingeniería en modelos de ida y vuelta manteniendo perfecta coherencia entre la modelación y el código, y permite además generación de código de alto nivel en un gran número de lenguajes.

### **Rational Rose Enterprise Edition**

Rational ofrece un Proceso Unificado (RUP) para el desarrollo de los proyectos de software, desde la etapa de Ingeniería de Requerimientos hasta la etapa de pruebas. Para cada una de estas etapas existe una herramienta que ayuda en la administración de los proyectos. Rose es la herramienta de Rational para la etapa de análisis y diseño de sistemas. Es una herramienta con plataforma independiente que ayuda a la comunicación entre los miembros del equipo, a monitorear el tiempo de desarrollo y a entender el entorno de los sistemas. Rose le permite a los diseñadores modelar sus componentes de forma individual y luego unirlos con otros componentes del proyecto. Además soporta

la construcción de componentes en lenguajes como C++, Visual Basic, Java, Ada, genera IDL's para aplicaciones CORBA.

### **2.10.1 ¿Por qué usar Rational Rose Enterprise Edition cómo herramienta de modelado?**

Después de hacer un estudio de las principales herramientas que soportan el Lenguaje de Modelado UML, se decidió emplear la herramienta informática de modelado Rational Rose Enterprise Edition, que está basada principalmente en el nivel de integración que tiene este con el resto de las herramientas que lo componen y por su capacidad de soportar diferentes plataformas. A continuación se describen algunas de las principales características de esta herramienta:

- Es en la actualidad, una de las herramientas CASE más potentes y es la que comercializan los desarrolladores de la Corporación Rational.
- Soporta de forma completa la especificación de UML. La cual permite a los arquitectos de software y desarrolladores visualizar el sistema completo utilizando un lenguaje común.
- Da un excelente soporte en el manejo de cambios durante el ciclo de vida del proyecto.

Además tiene un alto nivel de integración con el resto de las herramientas que lo acompañan en la Suite entre las que aparecen:

- Rational Clear CASE, para el control de versiones.
- Rational Clear Quest, para el control de cambios.
- Rational Model Integrator, para la integración de los artefactos.
- Rational Requisite Pro, herramienta de administración de requerimientos.

Brinda la posibilidad de generar y realizar ingeniería inversa en una buena cantidad de lenguajes de programación en su versión XDE. [(29)]

Permite que haya varias personas trabajando a la vez en el proceso iterativo controlado, para ello posibilita que cada desarrollador opere en un espacio de trabajo privado que contiene el modelo completo y tenga un control exclusivo sobre la propagación de los cambios en ese espacio de trabajo. Por todo lo anterior se puede decir que Rose es la herramienta de Análisis, Diseño, Modelado y Construcción de software Orientado a Objetos líder en el mercado.

### 2.11 HTTP Apache como servidor a utilizar

El Servidor HTTP Apache es un servidor HTTP de código abierto para plataformas Unix (BSD, GNU/Linux, etcétera), Windows y otras, que implementa el protocolo HTTP/1.1 y la noción de sitio virtual. El servidor Apache se desarrolla dentro del proyecto HTTP Server (httpd) de la Apache Software Foundation.

Entre sus principales características se encuentran:

- Funciona en plataformas virtuales.
- Apache es uno de los primeros servidores web en integrar el protocolo HTTP 1.1. Es totalmente compatible con el nuevo estándar HTTP 1.1.
- El servidor Apache no posee una interfaz de usuario gráfica para su administración. Se trata de un sencillo archivo de configuración llamado httpd.
- Soporte de host virtuales: Apache es además uno de los primeros servidores web en soportar tanto host basados en IP como host virtuales.
- Soporte de autenticación HTTP
- Soporte de scripts PHP.
- Servidor proxy integrado: puede convertir Apache en un servidor proxy cache.
- Estado del servidor y adaptación de registros: Apache le da una gran cantidad de flexibilidad en el registro y la monitorización del estado del servidor. El estado del servidor puede monitorizarse mediante un navegador web. Además, puede adaptar sus archivos de registro a su gusto. [ (30)]

### 2.12 Frameworks de PHP

En el desarrollo de software, un framework es una estructura de soporte definida en la cual otro proyecto de software puede ser organizado y desarrollado. Típicamente, un framework puede incluir soporte de programas, librerías y un lenguaje de scripting entre otros softwares para ayudar a desarrollar y unir los diferentes componentes de un proyecto.

Un framework representa una arquitectura de software que modela las relaciones generales de las entidades del dominio. Provee una estructura y una metodología de trabajo la cual extiende o utiliza las aplicaciones del dominio.

**Kumbia** está basado en software libre que pretende lograr una reducción sustancial en el tiempo de desarrollo de aplicaciones web sin que esto produzca efectos sobre los programadores.

Está basado en los siguientes conceptos:

- Compatible con muchas plataformas.
- Fácil de instalar y configurar.
- Fácil de aprender.
- Listo para aplicaciones comerciales.
- Convención sobre configuración.
- Simple en la mayor parte de casos pero flexible para adaptarse a casos más complejos.
- Soportar muchas características de aplicaciones web actuales.
- Soportar las prácticas y patrones de programación más productivos y eficientes.
- Producir aplicaciones fáciles de mantener.
- Basado en software libre.

Algunos de los principios de Kumbia es producir aplicaciones que sean prácticas para el usuario y no solamente para los programadores. Para ello automatiza todas las tareas de desarrollo repetitivas ayudando al programador a centrarse en la lógica de negocio de la aplicación y no en los pormenores que han de ser tenidos en cuenta durante el desarrollo de la misma. [(31)]

Kumbia es compatible con MySQL, Oracle y PostgreSQL y su modelo de objetos está diferenciado en tres capas:

- Abstracción de la base de datos
- Mapeo Objeto-Relacional
- Modelo MVC (Modelo, Vista, Controlador)

**Symfony** es un completo framework diseñado para optimizar el desarrollo de las aplicaciones web. Además, automatiza las tareas más comunes, permitiendo al desarrollador dedicarse por completo a los aspectos específicos de cada aplicación. El resultado de todas estas ventajas es que no se debe reinventar la rueda cada vez que se crea una nueva aplicación web.

Symfony está desarrollado completamente con PHP 5. Ha sido probado en numerosos proyectos reales y se utiliza en sitios web de comercio electrónico de primer nivel. Symfony es compatible con la mayoría de gestores de bases de datos, como MySQL, PostgreSQL, Oracle y SQL Server de Microsoft. Se puede ejecutar tanto en plataformas \*nix (Unix, Linux, etc.) como en plataformas Windows. [(32)]

Symfony automatiza la mayoría de elementos comunes de los proyectos web, como por ejemplo:

- La capa de internacionalización que incluye Symfony permite la traducción de los datos y de la interfaz, así como la adaptación local de los contenidos.

- La capa de presentación utiliza plantillas y layouts que pueden ser creados por diseñadores HTML sin ningún tipo de conocimiento del framework. Los helpers incluidos permiten minimizar el código utilizado en la presentación, ya que encapsulan grandes bloques de código en llamadas simples a funciones.
- Los formularios incluyen validación automatizada y relleno automático de datos, lo que asegura la obtención de datos correctos y mejora la experiencia de usuario.
- Los datos incluyen mecanismos de escape que permiten una mejor protección contra los ataques producidos por datos corruptos.
- La gestión de la caché reduce el ancho de banda utilizado y la carga del servidor.
- La autenticación y la gestión de credenciales simplifican la creación de secciones restringidas y la gestión de la seguridad de usuario.
- El sistema de enrutamiento y las URL limpias permiten considerar a las direcciones de las páginas como parte de la interfaz, además de estar optimizadas para los buscadores.
- El soporte de e-mail incluido y la gestión de APIs permiten a las aplicaciones web interactuar más allá de los navegadores.
- Los listados son más fáciles de utilizar debido a la paginación automatizada, el filtrado y la ordenación de datos.
- Los plugins, las factorías (patrón de diseño “Factory”) y los “mixin” permiten realizar extensiones a medida de Symfony.
- Las interacciones con Ajax son muy fáciles de implementar mediante los helpers que permiten encapsular los efectos JavaScript compatibles con todos los navegadores en una única línea de código.

### 2.12.1 ¿Por qué utilizar Symfony como framework PHP?

Luego de realizar un estudio de los diferentes framework se decidió utilizar Symfony como estructura de soporte para desarrollar y organizar el sistema, ya que este framework esta basado en software libre, separa la lógica de negocio, la lógica de servidor y la presentación de la aplicación web. Proporciona varias herramientas y clases encaminadas a reducir el tiempo de desarrollo de una aplicación web compleja. Además toma prestadas las mejores ideas de cualquier framework y las adapta a la suya.[(33)].

Esta diseñado para que se ajuste a los siguientes requisitos:

- Fácil de instalar y configurar.
- Independiente del sistema gestor de bases de datos.

- Sencillo de usar en la mayoría de los casos, pero lo suficientemente flexible como para adaptarse a los casos más complejos.
- Basado en la premisa de “convenir en vez de configurar”, en la que el desarrollador solo debe configurar aquello que no es convencional.
- Sigue la mayoría de mejores prácticas y patrones de diseño para la web.
- Preparado para aplicaciones empresariales y adaptables a las políticas y arquitecturas propias de cada empresa, además de ser lo suficientemente estable como para desarrollar aplicaciones a largo plazo.
- Código fácil de leer que incluye comentarios de phpDocumentor y que permite un mantenimiento muy sencillo.
- Fácil de extender, lo que permite su integración con librerías desarrolladas por terceros.

Además su arquitectura esta basada en el patrón arquitectónico que se va a utilizar en el sistema, el Modelo Vista Controlador (**MVC**). Symfony toma lo mejor de la arquitectura MVC y la implementa de forma que el desarrollo de aplicaciones sea rápido y sencillo, mediante la utilización de patrones de diseño se puede dividir la capa del modelo, en la capa de acceso a los datos y en la capa de abstracción de la base de datos.

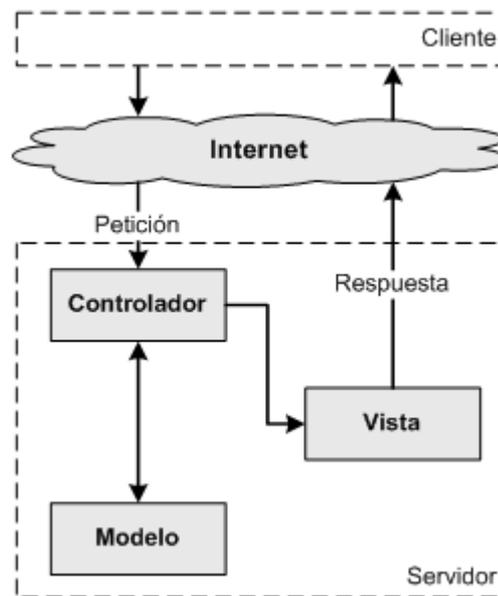
De esta forma, las funciones que acceden a los datos no utilizan sentencias ni consultas que dependen de una base de datos, sino que utilizan otras funciones para realizar las consultas. Así, si se cambia de sistema gestor de bases de datos, solamente es necesario actualizar la capa de abstracción de la base de datos. [(34)]

La arquitectura MVC proporciona grandes ventajas, como la organización del código, la reutilización, la flexibilidad y una programación mucho más entretenida. Por si fuera poco, crear la aplicación con Symfony permite crear páginas XHTML válidas, depurar fácilmente las aplicaciones, crear una configuración sencilla, la abstracción de la base de datos utilizada, enrutamiento con URL limpias, varios entornos de desarrollo y muchas otras utilidades para el desarrollo de aplicaciones.

El MVC esta formado por:

- El **modelo** representa la información con la que trabaja la aplicación, es decir, su lógica de negocio.
- La **vista** transforma el modelo en una página web que permite al usuario interactuar con ella.
- El **controlador** se encarga de procesar las interacciones del usuario y realiza los cambios apropiados en el modelo o en la vista.

La arquitectura MVC separa la lógica de negocio (el modelo) y la presentación (la vista) por lo que se consigue un mantenimiento más sencillo de las aplicaciones. Si por ejemplo una misma aplicación debe ejecutarse tanto en un navegador estándar como un navegador de un dispositivo móvil, solamente es necesario crear una vista nueva para cada dispositivo; manteniendo el controlador y el modelo original. El controlador se encarga de aislar al modelo y a la vista de los detalles del protocolo utilizado para las peticiones (HTTP, consola de comandos, e-mail, etc.). El modelo se encarga de la abstracción de la lógica relacionada con los datos, haciendo que la vista y las acciones sean independientes de, por ejemplo, el tipo de gestor de bases de datos utilizado por la aplicación.



**Fig. 2.1: El patrón MVC**

El principio más importante de la arquitectura MVC es la separación del código del programa en tres capas, dependiendo de su naturaleza. La lógica relacionada con los datos se incluye en el modelo, el código de la presentación en la vista y la lógica de la aplicación en el controlador.

Algunos de los beneficios del patrón MVC son:

- Múltiples vistas. La aplicación puede mostrar el estado del modelo en una variedad de formas y hacerlo de forma dinámica.
- Modular. El modelo y la vista están flojamente acoplados así que pueden cambiarse por completo de forma independiente sin necesidad de mucha reprogramación.
- Adecuado para el crecimiento. Se pueden añadir y actualizar controladores y vistas conforme cambia el modelo. [(35)].

### 2.13 Otras herramientas utilizadas

#### DreamWeaver

DreamWeaver es una herramienta con una gran flexibilidad y potencia en el diseño web mundialmente conocida.

- Permite realizar diseños perfectos en la vista Diseño de la forma más cómoda.
- Proporciona una potente combinación de herramientas visuales de diseño, funciones de desarrollo de aplicaciones y soporte para la edición del código.
- Su total compatibilidad con CSS facilitan la visualización, edición y desplazamiento dentro de los archivos, acelerando el flujo de compatibilidad con el navegador.

Una de las grandes ventajas que posee es, su compatibilidad con las principales tecnologías de desarrollo web, como son: HTML, XHTML, CSS, XML, JavaScript, Ajax, PHP entre otras.[ (36)].

#### Case Studio2

Se escogió CASE Studio 2 para modelar la base de datos por ser una herramienta altamente profesional y adaptable que permite a diseñadores y desarrolladores de base de datos crear y mantener visualmente diagramas de entidad relación (**DER**), diagramas de flujo de datos (**DFD**).

Tiene soporte para trabajar con una amplia variedad de formatos de base de datos (Oracle, SQL, MySQL, PostgreSQL, Access, etc.) y permite además generar Scripts SQL, aplicar procesos de ingeniería inversa a las bases de datos, usar plantillas de diseño personalizables, crear detallados informes en HTML y documentación de diagramas de flujo de datos exportables en formato XML.[(37)]

### 2.14 Conclusiones Parciales

En este capítulo luego de analizar las tendencias y tecnologías actuales, se llegó a la conclusión que se utilizará PHP como lenguaje de programación. Eclipse como IDE de PHP y Apache como servidor de la aplicación. Symfony como estructura de soporte para desarrollar y organizar el sistema y PostgreSQL como gestor de base de datos. RUP como metodología a seguir y UML como lenguaje gráfico para especificar, construir y documentar los artefactos del sistema. Se decidió además emplear como herramienta de modelado Rational Rose Enterprise Edition, y Case Studio para modelar la base de datos.

Después de seleccionar las diferentes herramientas que se emplearan para el desarrollo del sistema en el próximo capítulo se podrá modelar el negocio y el sistema de la aplicación.



# *CAPÍTULO 3*

## *MODELO DE NEGOCIO Y DE SISTEMA*

*“El estudioso es el que lleva a los demás a lo que él  
ha comprendido: la Verdad.”*

*Santo Tomás De Aquino*

### Introducción

En el capítulo que se desarrolla a continuación se realiza el Modelado del Negocio del control de los RRHH de la facultad 9 donde se definen los Actores, Trabajadores y Entidades del Negocio. Se identifican y modelan los Casos de Uso de la etapa del Negocio llegándose a identificar y plantear las Reglas esenciales del mismo. Se obtienen los requisitos funcionales y los no funcionales y la descripción del sistema propuesto.

### 3.1 ¿Que es un modelo?

Un modelo es una invención: algo que se concibe para explicar una serie de datos que se desea interpretar.

Para que un modelo sea útil, tiene que permitir que todos los datos "encajen" de forma coherente, es decir, tiene que poder explicar lo que pasa de una manera lógica. [(38)]

Por su propia definición, un modelo debe cumplir con tres requisitos básicos:

General, es decir, debe ser válido para cualquier aplicación del campo que formaliza.

- Abstracto, ya que con esto se pueden separar las características particulares del objeto de estudio para extraer su esencia.
- Consistente, para lograr que cada elemento tenga una única definición, acorde con la función que se espera que represente y coherente con el resto de componentes del modelo.

### Modelado de Negocio. ¿En qué consiste?

En los últimos años se han dedicado esfuerzos a la búsqueda de procesos de producción de software que sean precisos, rigurosos y confiables. La mayoría de estas investigaciones tienen como punto de partida los requisitos del sistema de software. Generalmente estas investigaciones restan importancia a los Modelos de Negocio, los cuales permiten visualizar la forma de operar de la empresa, así como las necesidades de los usuarios del sistema a desarrollar. [(38)]

En un proceso de producción de software que no tenga como primera etapa el modelado de procesos de negocios, cualquier esfuerzo para obtener los requisitos del sistema de información estará disminuido por la incapacidad de asegurar, la utilidad real de éste en el contexto de las tareas organizacionales. [(38)]

El Modelado de Negocios puede ser definido como la abstracción de los elementos de una organización y las relaciones entre ellos y como una forma organizacional lateral u horizontal que encapsula las interdependencias de tareas, roles, recursos humanos, departamentos y funciones

requeridas para proveer un producto o servicio a un cliente. Definen la dinámica del comportamiento del entorno, actuando sobre entidades o recursos. [(26; 38)]

### **3.2 Modelo del Negocio del Subsistema de RRHH**

Para modelar el proceso de negocio, primeramente se seleccionan los actores y trabajadores que intervienen en el negocio, luego se definen los casos de usos que lo integran conformando el diagrama de casos de uso del negocio. Por último con los trabajadores y los objetos utilizados en el proceso se confecciona el modelo de objetos del negocio, cumpliendo así con la salida de todos los artefactos del flujo de trabajo.

#### **3.2.1 Actores y Trabajadores del Negocio**

##### **Actores del Negocio**

Como se había indicado en epígrafes anteriores, en la primera etapa que describe RUP para el desarrollo de software, se deben identificar los actores del mismo, siendo concebidos estos como el individuo, grupo, organización, máquina o sistema que interactúa y se beneficia con la existencia del negocio.

Ejemplo:

- Clientes.
- Socios.
- Proveedores.
- Autoridades.
- Propietarios. Gerentes, Administradores.
- Sistemas de información externos al negocio.

Actores del Subsistema	Justificación
<b>RRHH</b>	
Profesor	Persona que labora en la docencia, producción e investigación de la facultad.
Trabajador	Persona que realiza diferentes tareas en la facultad.

### **Trabajadores del Negocio.**

Representan a personas, o sistemas (software) dentro del negocio, son los que ejecutan los procesos involucrados en el negocio.

Estos trabajadores están dentro de la frontera del negocio, son los que en un futuro se convertirán en usuarios del sistema a construir.

Cada trabajador del negocio debe definirse brevemente con su responsabilidad dentro del negocio.

[(26)]

Trabajadores del Subsistema RRHH	Justificación
Asistente de Control	Persona responsable del control y registro de los profesores y trabajadores de la facultad.

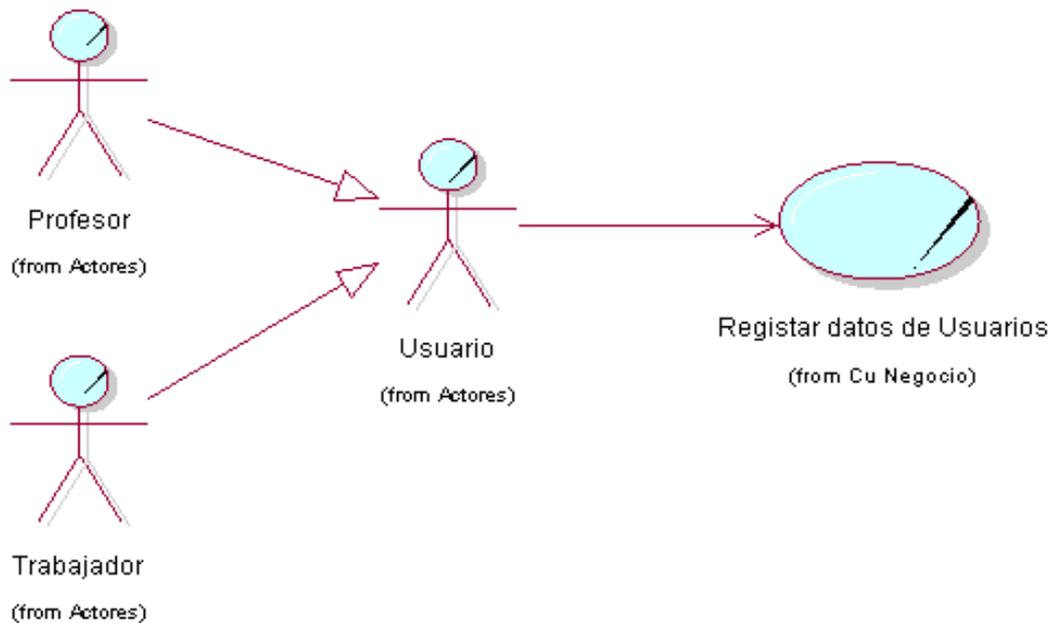
### **3.2.2 Proceso de Negocio del Subsistema de RRHH**

En el Modelo de Negocio del control de los RRHH de la facultad 9 existe un proceso fundamental que es el Registro de Usuarios.

Este proceso se realiza cuando un nuevo profesor o trabajador es asignado a la facultad. El mismo se presenta en la secretaría de la Facultad para que sus datos sean registrados.

### **3.2.3 Diagramas de Casos de Uso del Subsistema de RRHH**

El Modelo de Casos de Uso del Negocio representa un sistema (en este caso, el negocio) desde la perspectiva de su uso, y esquematiza cómo proporciona valor a sus usuarios (en este caso, sus clientes y socios). [(26)]



**Fig. 3.1: Diagrama de CU del Negocio**

Los Diagramas de Casos de Uso del Negocio se realizan con el fin de tener una visión general de los diferentes procesos de negocio de la organización. En este tipo de diagrama aparece cada proceso del negocio como un Caso de Uso. Este diagrama permite mostrar los límites y el entorno de la organización bajo estudio. [(39)]

### 3.2.4 Descripción Textual de los Casos de Uso del Negocio

En este nuevo epígrafe se desarrollan los Casos de Uso del Negocio, es decir se describen la secuencia de pasos de las operaciones que se efectúan en la secretaría.

<b>Caso de Uso:</b>	<b>Registrar datos de Usuarios.</b>
<b>Actores:</b>	Profesor, Trabajador.
<b>Trabajadores:</b>	Asistente de Control
<b>Resumen:</b>	El caso de uso se inicia cuando el profesor o trabajador va a la secretaría para oficializar su ingreso a la facultad. La Asistente de Control solicita los datos, y los registra.
<b>Precondiciones:</b>	Que en Capital humano le hayan dado la ubicación para la

Facultad 9	
Acción del Actor.	
Respuesta del Negocio.	
1. El profesor o trabajador solicita su registro.	1.1. La Asistente de Control le solicita los datos.
2. El profesor o trabajador brindan los datos.	2.1. La Asistente de Control llena el formulario de registro.
Poscondiciones	Se tiene el registro de los profesores y trabajadores de la facultad.

### 3.3 Modelo de Objetos del Subsistema de RRHH

Un Modelo de Objetos es uno de los artefactos que define RUP a obtener en esta etapa del Negocio. Este modelo, representa la relación que existe entre los trabajadores del entorno para el que se desarrolla el software y cómo estos laboran o utilizan las entidades del mismo.

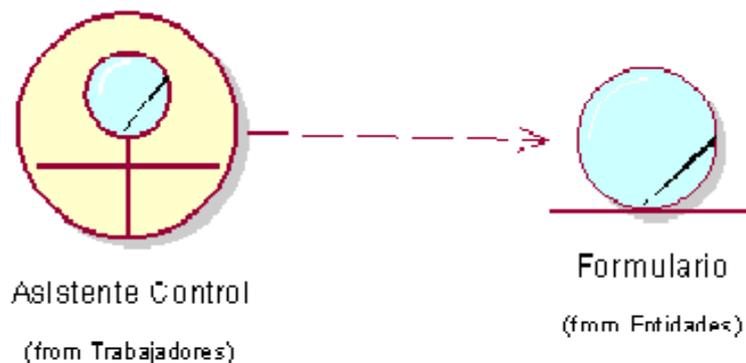


Fig. 3.2: Modelo de Objetos

### 3.4 Reglas del Negocio

Las Reglas del Negocio son los parámetros o restricciones por los que este se rige de modo que no sea posible realizar acciones no válidas. En el caso del subsistema de RRHH deben ser de carácter obligatorio para el correcto cumplimiento del subsistema.

- La asistente de control es la responsable de introducir eliminar y modificar los datos de los profesores y trabajadores de la facultad.

### 3.5 Requerimientos Funcionales de la Plataforma

Luego de obtener una visión de la organización, los posibles procesos para automatizar, roles y responsabilidades de la organización en los modelos de casos de uso del negocio y de objetos fueron identificados los siguientes requisitos funcionales para el desarrollo del sistema propuesto:

#### R1- Autenticar Usuario.

- 1.1- Verificar que el usuario este registrado en el sistema.
- 1.2- Verificar usuario y contraseña en el dominio UCI.

#### R2- Gestionar Datos

- 2.1- El sistema debe permitir registrar datos.
- 2.2- El sistema debe ser capaz de eliminar datos.
- 2.3- El sistema debe permitir la modificación de los datos registrados.

R3-El sistema debe generar reportes según criterios especificados, díganse reportes de profesores por departamento, por categoría docente, por categoría científica entre otros.

R4-El sistema debe permitir imprimir reportes.

R6-El sistema debe de ser capaz de buscar los datos de los usuarios que se encuentren en la base de datos, basándose en diferentes criterios de búsqueda.

R7- Gestionar las noticias que se publican en la intranet.

- 7.1- El sistema debe permitir insertar noticias.
- 7.2- El sistema debe ser capaz de eliminar noticias.
- 7.3- El sistema debe permitir la modificación de las noticias.

R8- El sistema debe permitir visualizar a los usuarios las noticias que se publican en la intranet.

R9- El sistema de permitir buscar noticias, que hayan sido publicadas en la intranet, por el título.

### 3.6 Requerimientos No Funcionales de la Plataforma

Los requerimientos no funcionales son cualidades o propiedades que el producto debe tener. Debe pensarse en estas propiedades como las características que hacen al producto atractivo, usable, rápido o confiable. Los requerimientos no funcionales del sistemas son fundamentales para el buen funcionamiento del sistema, aquí radica su importancia.

A continuación se presentan los requerimientos no funcionales del sistema:

### **Requerimientos de software**

- Para la implantación del sistema se requiere de un servidor Linux ó Windows NT.
- Apache como servidor web y PostgreSQL-8 como servidor de bases de datos.
- Los requerimientos en el lado del cliente para la utilización del sistema sólo se limitan a tener disponible un navegador web compatible o superior con Internet Explorer.

### **Requerimientos de hardware**

- La máquina servidora debe tener como mínimo las siguientes características de hardware: Procesador Pentium 3.00 GHz o superior, 512 MB de memoria RAM como mínimo (incluye la utilizada por el SO) y 80Gb de capacidad en disco duro.
- Las computadoras situadas en los puestos de trabajo de los usuarios requieren como mínimo un procesador Pentium 281 GHz, 240 MB de memoria RAM como mínimo. Estas máquinas deben de estar conectadas en red con el servidor.

### **Apariencia o interfaz externa**

- El software debe brindar una interfaz simple y de fácil uso para que el usuario no tenga dificultad al interactuar con el sistema.
- El sistema debe estar diseñado de modo tal que el usuario pueda ir de un punto a otro dentro del mismo con gran facilidad.
- La aplicación debe ser lo más interactiva posible.

### **Usabilidad**

- El sistema podrá ser usado por los usuarios de la facultad 9.

### **Rendimiento**

- Se debe garantizar que el tiempo de respuesta del sistema ante las solicitudes de los usuarios para cada acción a realizar por estos sea el menor posible, para garantizar el buen funcionamiento del sistema debido a que es muy dinámico e interactivo.

### **Portabilidad**

- El sistema se ejecutará sobre Windows y Linux, permitiendo que el servidor pueda ser cambiado sin importar el sistema operativo, llevando a cabo esta acción sin necesidad de efectuar cambios significativos.

**Requerimientos de Seguridad**

- Para mantener la integridad en el mismo sólo se podrá acceder al sistema después de autenticarse.

**Requerimientos de confiabilidad**

- Además de ser seguro es muy confiable ya que garantiza que los datos lleguen íntegros e intactos a su destino. Los datos son almacenados en una base de datos segura.

**3.7 Modelo de Sistema**

El sistema que se modela para la plataforma incluye la intranet y el subsistema RRHH. El modelo de sistema propuesto tiene siete casos de uso: Gestionar, Buscar y Visualizar Noticias para la intranet, Gestionar Datos, Generar Reportes y Buscar Usuarios para el subsistema RRHH y Autenticar Usuario que se utiliza en todos los subsistemas que componen la plataforma.

**3.7.1 Descripción de los actores de la plataforma**

Los actores de la plataforma son todos los que intervienen en cada uno de los subsistemas.

Actores del Sistema	Justificación
Asistente de Control	<p>Persona responsable del control y registro de los profesores y trabajadores de la facultad.</p> <p>Además se encarga de la entrega del módulo docente a cada uno de los profesores vinculados a la docencia.</p>
Jefe Departamento	<p>Es la máxima autoridad en un Departamento Docente. Realiza las operaciones de Gestionar el Plan de Trabajo del Departamento, es el encargado de evaluar a los Jefes de Disciplina de su departamento, evalúa las actividades de los Planes de Trabajo del Departamento y de las</p>

	<p>Disciplinas correspondientes a su departamento, además puede generar el por ciento de cumplimiento de cualquier plan de trabajo del Departamento. Aprueba el Plan de Trabajo del Departamento y los Planes de Trabajos de las Disciplinas correspondientes a su Departamento. Es el encargado de informar las afectaciones de sus respectivos profesores, así como aprobar o no los cambio de turnos solicitados por estos.</p> <p>Es el responsable de llevar un control estricto del desempeño de los trabajadores de su área, así como comunicar la evaluación de cada uno de ellos.</p>
Jefe Disciplina	<p>Realiza las operaciones de Gestionar el Plan de Trabajo de la Disciplina, es el encargado de evaluar a los Jefes Asignaturas y de evaluar las actividades de los Planes de Trabajo de las Asignaturas. Aprueba los Planes de Trabajo de las asignaturas que pertenecen a su Disciplina.</p>
Jefe Asignatura	<p>Realiza las operaciones de Gestionar el Plan de Trabajo de la asignatura. Es el encargado de evaluar a los Profesores que imparten su Asignatura y evalúa las actividades de los Planes de Trabajo Individual de los Profesores de su Asignatura. Es el encargado de Gestionar el Programa Analítico de la asignatura.</p>
Profesor	<p>Realiza las operaciones de Gestionar su Plan de Trabajo Individual.</p>

Planificadora Docente	Se encarga de controlar las afectaciones de los profesores, de realizar los cambios de turnos y facilitar locales solicitados por estos mismos.
Secretaria Docente	Es la responsable de controlar y registrar toda la información relacionada con la facultad.
TECADIES	Es la encargada de realizar la ratificación de matrícula de cada estudiante al inicio de cada curso y de apoyar a la Secretaria Docente en las actividades que esta realiza.
Dirección Producción FAC9	Es la máxima autoridad en un Departamento de Producción. Realiza la Gestión de la Producción en la facultad, es el encargado de crear un polo, proyecto, y aunque no es su responsabilidad puede crear EPI, Y Líneas de investigación así como elegir sus jefes. Además es el responsable de elegir el jefe de polo, evaluar el desempeño de todos los profesores, definir los cursos optativos, asignar tareas y llevar el control de todos los eventos en los que participa cada estudiante.
Directivo Polo	Es la máxima autoridad dentro del polo productivo, es el encargado de seleccionar los jefes de proyecto, tutores de EPI, y realizar los diagnósticos a los estudiantes para determinar en la fase que se encuentra. Puede además aunque no es su responsabilidad responsable evaluar

	desempeño en un momento determinado del estudiante tanto de proyecto como de línea de investigación.
Directivo Línea de investigación	Es la máxima autoridad dentro de la línea de investigación, es el encargado de evaluar el desempeño de los estudiantes que trabajan en la línea.
Directivo Proyecto	Es la máxima autoridad dentro del proyecto, es el encargado de evaluar el desempeño de los estudiantes que integran el proyecto.
Tutor EPI	Es la máxima autoridad dentro del EPI, encargado de gestionar las líneas de investigación y puede realizar las mismas tareas que los directivos de líneas de investigación y proyecto.
Estudiante	Es el encargado de escoger los polos en lo que quiere estar.
Vicedecano EUR	Es el máximo responsable de controlar, y centrar la información de todos los profesores internos y de los estudiantes en la residencia, forma de localizarlos, también de controlar las evaluaciones en todas las tareas extracurriculares, tener el control y actualizar la distribución de los mismos en la Residencia. Es el encargado de evaluar a los profesores en otras tareas como Guardia Obrera y Visitas a la Residencia.

Personal del Vicedecanato de Residencia	Son los encargados de revisar la realización de las tareas, controlan y reúnen toda la información, pueden registrar, modificar, mostrar las evaluaciones, así como hacer movimientos en la Residencia autorizados por el Vicedecano.
Instructor Educativo	Es el encargado de pasar inspección y evaluar a los estudiantes y profesores internos, controlando la limpieza, la organización. Evalúa la realización de la cuartelaría por parte de los estudiantes, entrega el módulo de aseo personal, así como reporta los problemas constructivos e incidencias que ocurren en los apartamentos que atiende.
Usuario General	Este actor permite la autenticación a los actores de los diferentes subsistemas y al administrador.
Administrador de la plataforma	Este actor es el encargado de publicar las noticias en la intranet, así como actualizar la base de datos utilizando los web service.
Secretaria de Confección de Nómina	Es la responsable de recoger todos los datos de los distintos departamentos y confeccionar la pre nómina de pago.
Decano	Encargado de solicitar reportes relacionados con el control de Nómina, Evaluación del Desempeño y Entrega de Materiales a los trabajadores de la facultad 9.

### 3.7.2 Casos de Uso de la plataforma

- Autenticar Usuario.

- Gestionar Datos.
- Generar Reporte.
- Buscar Usuarios.
- Gestionar Noticias.
- Visualizar Noticias.
- Buscar Noticia.

### 3.7.3 Descripción de los Casos de Uso de la plataforma

Caso de Uso	
01	Autenticar Usuario
Propósito	Verificar que no accedan usuarios sin privilegios adecuados a los diferentes módulos.
Actor	Usuario General
Resumen	El caso de uso se inicia cuando el Usuario General introduce su usuario y contraseña y el sistema verifica a que módulos puede entrar.
Referencias	R1.
Precondiciones	
Flujo Normal de Eventos	



Acción del Actor	Respuesta del Sistema
1. El Usuario General se autentica en la página inicial de la intranet.	1.1. El sistema verifica que el usuario esté en la base de datos. 1.2. El sistema utilizando el protocolo Ldap verifica en el servidor de dominio uci si son correctos el usuario y la contraseña.

	1.3. El sistema verifica que tipo de usuario es: profesor o trabajador y busca en la base de datos el cargo que ocupa el usuario.
2. El Usuario General selecciona el módulo al que quiere acceder.	2.1. Según el cargo el sistema establece las acciones que el usuario puede realizar en el módulo.
Curso alternativo de los eventos	
3.1. Si el usuario no se autentica y quiere acceder a un módulo el sistema mostrará el mensaje: "Debe autenticarse en el sistema con su usuario y contraseña" y se vuelve a mostrar la página de inicio.	
3.2. Si el usuario no se encuentra en la base de datos o la contraseña es incorrecta el sistema mostrará el mensaje: " Ese usuario no está en la base de datos".	
3.3. Si el usuario no tiene acceso al módulo seleccionado por él, se muestra un mensaje: "Su credencial no tiene permiso para acceder a este módulo".	
Prioridad	Crítico.

Caso de Uso	
02	Gestionar Datos.
Propósito	Controlar los datos de los trabajadores y profesores.
Actor	Asistente de Control
Resumen	El caso de uso se inicia cuando la secretaria introduce, modifica, elimina los datos referentes a los profesores o trabajadores de la facultad 9.
Referencias	R2.
Precondiciones	En caso de eliminar y modificar los datos, deben primeramente existir.
Curso Normal de Eventos	
Acción del Actor	Respuesta del Sistema
1. La Asistente de Control selecciona que desea hacer con los datos:	1.1- a) Si desea insertar un nuevo usuario, ver sección: Registrar Usuario

- b) Si desea modificar una orden, ver sección: Modificar Usuario.
- c) Si desea eliminar una orden, ver sección: Eliminar Usuario.

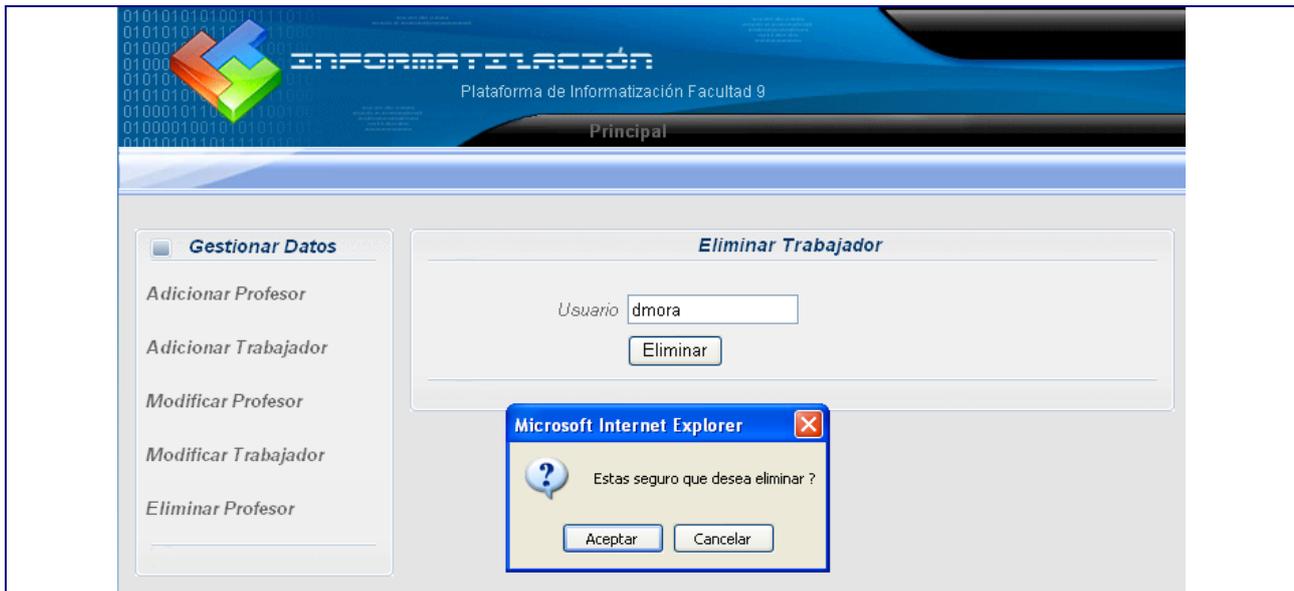
**Sección “Registrar Usuario”**



Acción del Actor	Respuesta del Sistema
2. La Asistente de Control busca el usuario a quien desea registrar nuevos datos.	2.1. El sistema muestra el formulario de registro con los datos que se cargan de los servicios web.
3. La Asistente de Control inserta los nuevos datos.	3.1. El sistema verifica los datos.

**Sección “Modificar Usuario”**

Acción del Actor	Respuesta del Sistema
4. La Asistente de Control busca el usuario que desea modificar.	4.1. El sistema presenta los datos registrados del usuario seleccionado.
5. La Asistente de Control especifica los datos que desea cambiar.	5.1. El sistema valida los datos. 5.2. Si son correctos guarda los cambios en la base de datos.
<b>Sección “Eliminar Usuario”</b>	



Acción del Actor	Respuesta del Sistema
6. La Asistente de Control introduce el usuario que desea eliminar.	6.1. El sistema pide confirmación de eliminación.
9. La Asistente de Control confirma la eliminación del usuario.	9.1. El sistema elimina el usuario seleccionado.
Curso alternativo de los eventos	
Si se quiere registrar, modificar o eliminar un trabajador y el usuario que se introduce para buscar en el sistema es el de un profesor se mostrará el mensaje: "Ese usuario no es trabajador". Para el caso contrario el sistema muestra el siguiente mensaje: "Ese usuario no es profesor".	
Prioridad	Crítico.

Caso de Uso	
03	Generar Reporte
Propósito	Obtener reportes específicos en dependencia de los datos.
Actor	Asistente de Control.
Resumen	El caso de uso se inicia cuando la Asistente de Control introduce los datos específicos. El sistema busca en la base datos y generará un reporte.

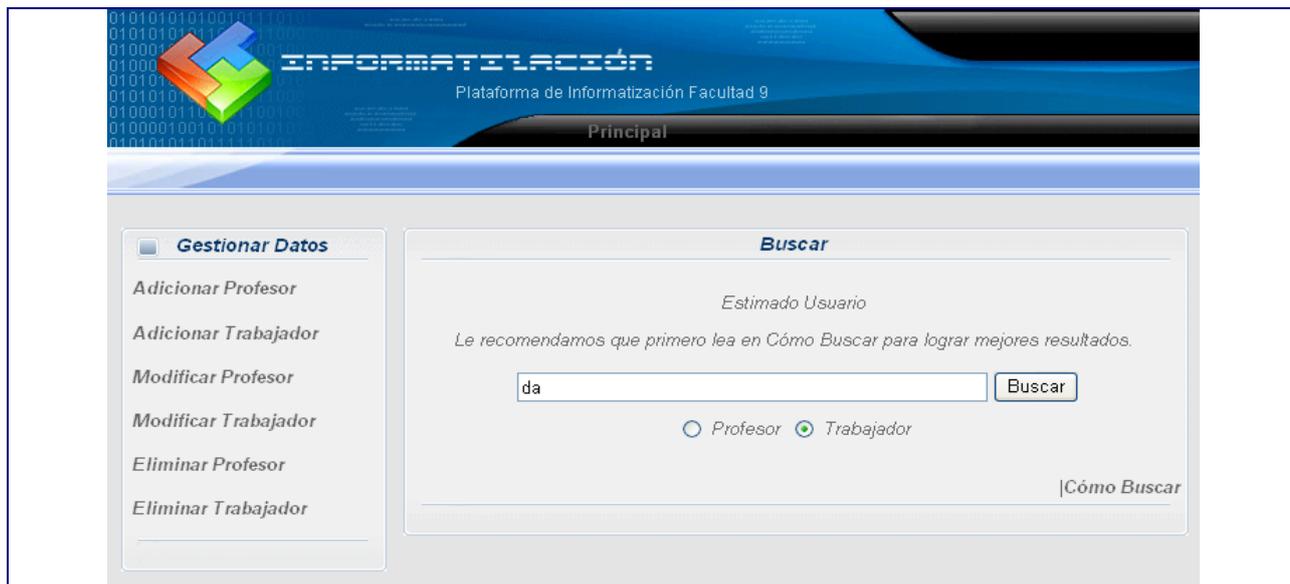
Referencias	R3, R4, R5.
Precondiciones	Tener usuarios registrados en la Base de Datos.

Reporte de Profesores por Departamento Ciencias Basicas			
Nombre y Apellidos	Usuario	Solapin	Carnet de Identidad
Yoient Ramos Lao	yramos	55339	888
Pedro Valdes Rodriguez	pvaldes	2222	5566

<b>Curso Normal de Eventos</b>	
Acción del Actor	Respuesta del Sistema

1. La Asistente de Control especifica los datos de entrada para generar reporte.	<p>1.1.La aplicación hace la búsqueda en la Base de Datos</p> <p>1.3. La aplicación genera el reporte.</p> <p>1.3.1. Si desea graficar el reporte ir a sección: Graficar Reporte.</p> <p>1.3.2. Si desea imprimir el reporte ya sea graficado o no ir a sección: Imprimir Reporte</p>
<b>Sección Imprimir Reporte</b>	
Acción del Actor	Respuesta del Sistema
2. La Asistente de Control solicita imprimir un reporte.	2.1. El sistema le brinda la opción para imprimir.
Prioridad	Crítico.

Caso de Uso	
04	Buscar Usuarios
Propósito	Buscar información sobre los usuarios que aparezcan en la base de datos.
Actor	Asistente de Control
Resumen	El caso de uso se inicia cuando la Asistente de Control introduce el criterio de búsqueda como: el nombre, primer apellido, segundo apellido, usuario y solapín.
Referencias	R6
Precondiciones	Que exista al menos un profesor o un trabajador en la base de datos.



Flujo Normal de Eventos	
Acción del Actor	Respuesta del Sistema
1. La Asistente de Control introduce el dato de búsqueda.	1.1. El sistema hace la búsqueda en la base de datos. 1.3. El sistema muestra la información sobre el usuario.
Prioridad	Crítico.

Caso de Uso	
05	Gestionar Noticias
Propósito	Controlar la inserción, edición y eliminación de las noticias en la intranet.
Actor	Administrador Plataforma
Resumen	El caso de uso se inicia cuando el administrador introduce, modifica, elimina noticias de la intranet.
Referencias	R7.
Precondiciones	En caso de eliminar y modificar deben existir noticias en la base datos
Curso Normal de Eventos	
Acción del Actor	Respuesta del Sistema
1. El administrador de la plataforma selecciona la acción a realizar con las noticias.	1.1- a) Si desea insertar una nueva noticia, ver sección: Insertar Noticia b) Si desea modificar noticia, ver sección: Modificar Noticia. c) Si desea eliminar una orden, ver sección: Eliminar Noticia.
<b>Sección "Insertar Noticia"</b>	

**Insertar Noticia**

*Título*

*Autor*

*Descripción*

*Resumen*

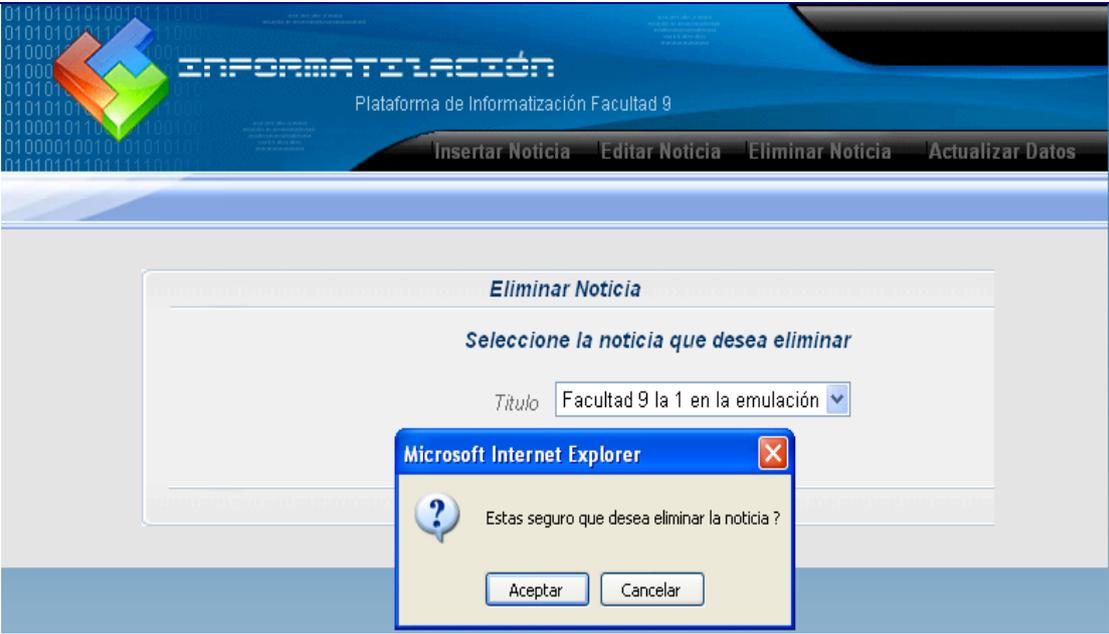
*Fecha\_Publicacion*  
 ...

*Tipo de Noticia*

Acción del Actor	Respuesta del Sistema
2. El administrador plataforma solicita insertar un nueva noticia.	2.1. El sistema muestra el formulario registro correspondiente.
3. El administrador plataforma inserta la nueva noticia.	3.1. El sistema muestra la noticia.
<b>Sección “Editar Noticia”</b>	



Acción del Actor	Respuesta del Sistema
4. El administrador plataforma selecciona la noticia que desea modificar.	4.1. El sistema presenta los datos registrados de de la noticia seleccionada.
5. El administrador plataforma modifica los datos que desea cambiar en el formulario	5.1. El sistema guarda los cambios realizados a la noticia.

Sección “Eliminar Noticia”	
	
Acción del Actor	Respuesta del Sistema
6. El administrador plataforma selecciona la noticia que desea eliminar.	6.1. El sistema pide confirmación de eliminación.
9 El administrador plataforma confirma la eliminación de la noticia.	9.1. El sistema elimina la noticia seleccionada.
Curso alternativo de los eventos	
Prioridad	Crítico.

Caso de Uso	
06	Visualizar Noticias
Propósito	Permitir que los usuarios visualicen las noticias publicadas en la intranet.
Actor	Usuario General.
Resumen	El caso de uso se inicia cuando el Usuario General selecciona el título de la noticia que quiere leer y el sistema le muestra la noticia.
Referencias	R8

Precondiciones	
<p style="text-align: center;"><i>Destacan legado épico del Che</i></p>  <p>Carlos Lage Dávila, vicepresidente del Consejo de Estado de la República de Cuba, afirmó hoy que el legado épico de Ernesto Che Guevara cobra dimensiones superiores en la nueva realidad que viven los pueblos. En gala homenaje por el aniversario 80 del natalicio del Guerrillero Heroico, presidida por el General de Ejército Raúl Castro Ruz, Presidente de los Consejos de Estado y de Ministros, y que tuvo por sede el teatro Karl Marx, Lage subrayó la coincidencia histórica con el nacimiento de Antonio Maceo, quien con la Protesta de Baraguá marcó el devenir ineludible de los cubanos de ley. Expresión del alto nivel artístico del tributo fue el estreno de EL cantar del caballero y su destino, obra del maestro José María Vitier inspirada en la vida y el pensamiento del Che y su impacto latinoamericano, africano y universal, escrita para orquesta sinfónica y dedicada al hombre que se consagra a la lucha por la justicia y el mejoramiento humano. Entre otros artistas, Vitier dirigió brillantemente en la interpretación de su obra a la Orquesta Sinfónica Nacional, los Coros Nacional y Exaudi y a la soprano Bárbara Yáñez, y conmovió al público con esta actuación que perdurará también por sus aportes artísticos y la organización del merecido homenaje al Che. HOMENAJES AL CHE EN ROSARIO Y VALLEGRANDE Acciones de homenaje al Comandante Ernesto Guevara se multiplican por el mundo. En su natal ciudad argentina de Rosario, los pobladores auspiciaron el encuentro internacional de Cátedras del Che, y en el Teatro El Círculo se desarrolló un panel académico a cargo del escritor y poeta cubano Miguel Barnet y el Premio Nobel de la Paz 1980, el argentino Adolfo Pérez Esquivel. La delegación cubana a los actos por el aniversario 80 del natalicio del Che la encabeza Rogelio Acevedo, presidente del Instituto de Aeronáutica Civil de la Isla e integrante de la columna guerrillera que el Che dirigió en la Sierra Maestra, hasta el Occidente de la Mayor de las Antillas. Muy emotiva resultó una sesión especial de la Cámara de Diputados de Santa Fe, en la cual estuvieron presentes las hijas del Guerrillero Heroico, Aleida y Celia Guevara March. En tanto, en la localidad boliviana de Vallegrande, un grupo de colaboradores cubanos recibirá un reconocimiento por su destacada labor en el país andino. Ese constituirá el preámbulo del acto central, que tendrá lugar en La Higuera, donde fue asesinado el legendario combatiente el 9 de octubre de 1967. RECORDARÁN EN CANTO A MACEO Y AL CHE El tributo de los pueblos hermanos de la Isla de Cuba, Antonio Maceo y Ernesto Guevara...</p>	
Flujo Normal de Eventos	
Acción del Actor	Respuesta del Sistema
1. El Usuario General selecciona el título de la noticia que desea.	1.1. El sistema muestra en otra página la noticia completa.
Prioridad	Crítico.

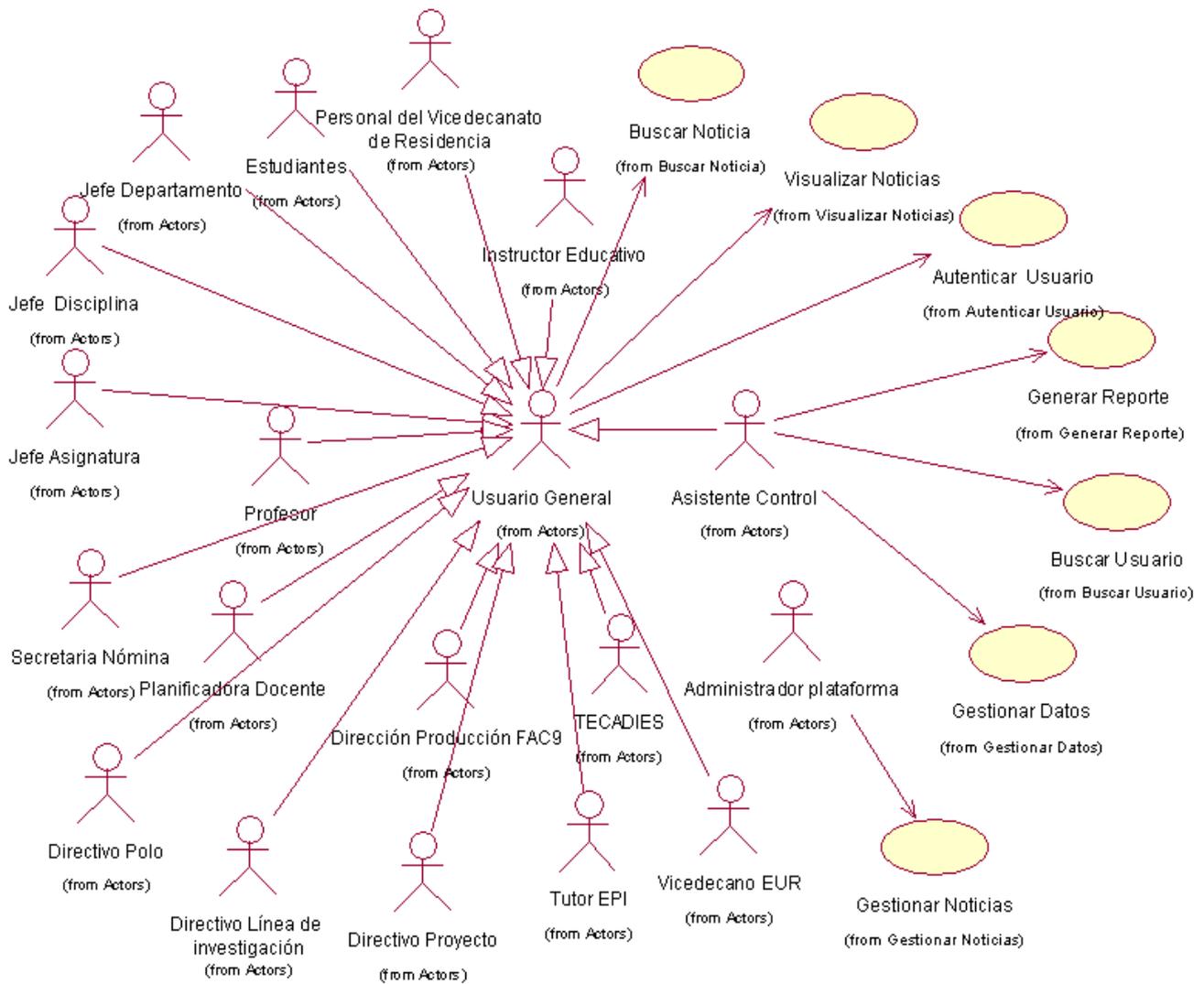
Caso de Uso	
04	Buscar Noticia
Propósito	Buscar las noticias asociadas a un título..
Actor	Asistente de Control
Resumen	El caso de uso se inicia cuando el Usuario General introduce una palabra o el título completo asociado a la noticia que se quiera buscar.
Referencias	R9
Precondiciones	Que existan noticias en la base de datos.



Flujo Normal de Eventos	
Acción del Actor	Respuesta del Sistema
1. El Usuario General introduce el dato de búsqueda.	1.1. El sistema hace la búsqueda en la tabla de Noticias 1.3. El sistema muestra la(s) noticia(s) encontrada(s) asociada(s) al dato de búsqueda.
Curso alternativo de los eventos	
3.2. Si no existe ninguna noticia en la base de datos que contenga en el título la cadena entrada por el usuario se muestra un mensaje informándole la situación al usuario.	
Prioridad	Crítico.

### 3.7.4 Diagrama de Caso de Uso de la Plataforma

Un diagrama de casos de uso es una representación gráfica de parte o el total de los actores y casos de uso del sistema, incluyendo sus interacciones. Todo sistema tiene como mínimo un diagrama, que es una representación gráfica del entorno del sistema y su funcionalidad principal.



**Fig. 3.3: Diagrama de CUS de la Plataforma.**

Un diagrama de casos de uso muestra, por tanto, los distintos requisitos funcionales que se esperan de una aplicación o sistema y cómo se relaciona con su entorno.

### **3.8 Conclusiones Parciales**

En este capítulo se llevo a cabo el modelamiento del negocio para la comprensión del problema a resolver. Se comenzó por un análisis del proceso a automatizar y la información que manejaba para poder comprender el negocio de la investigación. Se analizaron los requerimientos funcionales y no funcionales del sistema, respetando las solicitudes y necesidades de los clientes. Por último se diseñó el modelo del sistema donde se plantean todas las funcionalidades que el sistema debe cumplir.



## *CAPÍTULO 4*

### *CONSTRUCCION DEL SISTEMA*

*“En teoría, no existe diferencia entre teoría y práctica;  
en la práctica sí la hay.”*

*Jan L.A. van de Snepscheut*

## Capítulo 4. Construcción del Sistema

### Introducción

El objetivo principal del análisis y diseño en el desarrollo de una aplicación es transformar los requerimientos a una especificación que describa cómo implementar el sistema. El análisis fundamentalmente consiste en obtener una visión que se preocupa de ver que hace el sistema de software a desarrollar, por tal motivo este se interesa en los requerimientos funcionales. Por otro lado, el diseño es un refinamiento que toma en cuenta los requerimientos no funcionales, por lo cual se centra en como el sistema cumple sus objetivos.

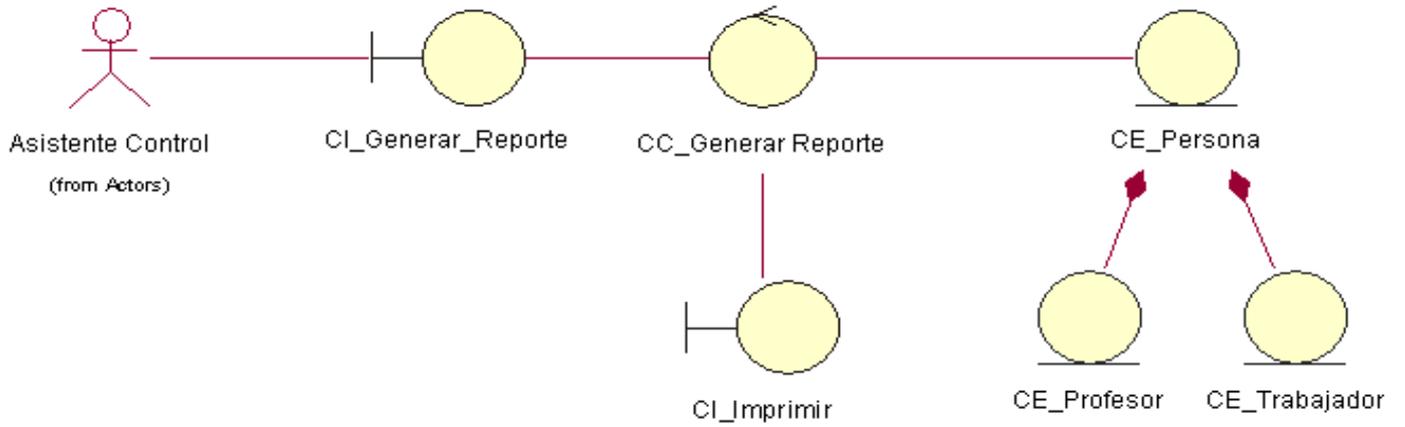
En el presente capítulo se realiza el análisis del sistema, modelando el diagrama de análisis, también se analiza las iniciaciones del diseño, donde se hace referencia a los estándares de interfaz, se fundamenta los tratamientos de errores. Para finalizar se realiza todo el proceso de diseño de la aplicación, donde se desarrolla el diagrama de clases de diseño, el modelo de datos y el modelo de despliegue.

### 4.1 Análisis de la intranet y el subsistema RRHH

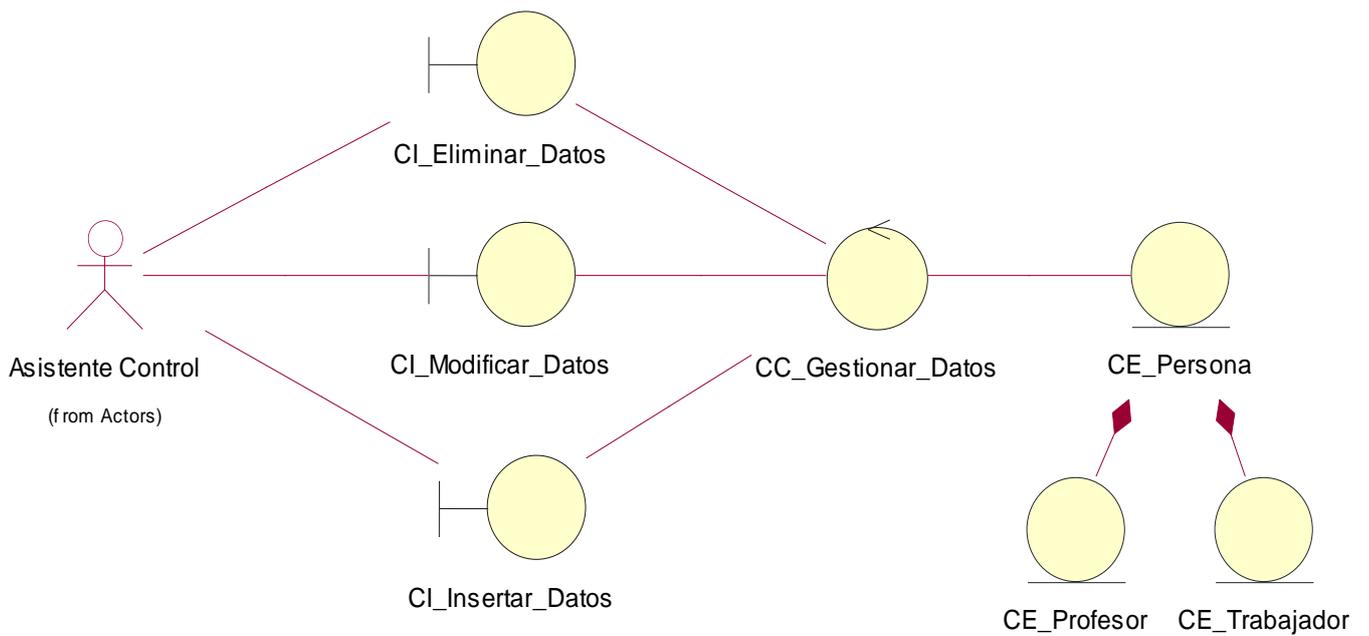
El análisis consiste en obtener una visión del sistema que se preocupa por ver ¿Qué hace?, de modo que sólo se interesa por los requisitos funcionales, y transformarlos en un diseño de clases viendo las relaciones e interacción que existe entre ellos, y teniendo en cuenta en el proceso una arquitectura fuerte que permita adaptar el sistema al entorno de implementación que se está desarrollando

#### 4.1.1 Diagrama de Clases del Análisis

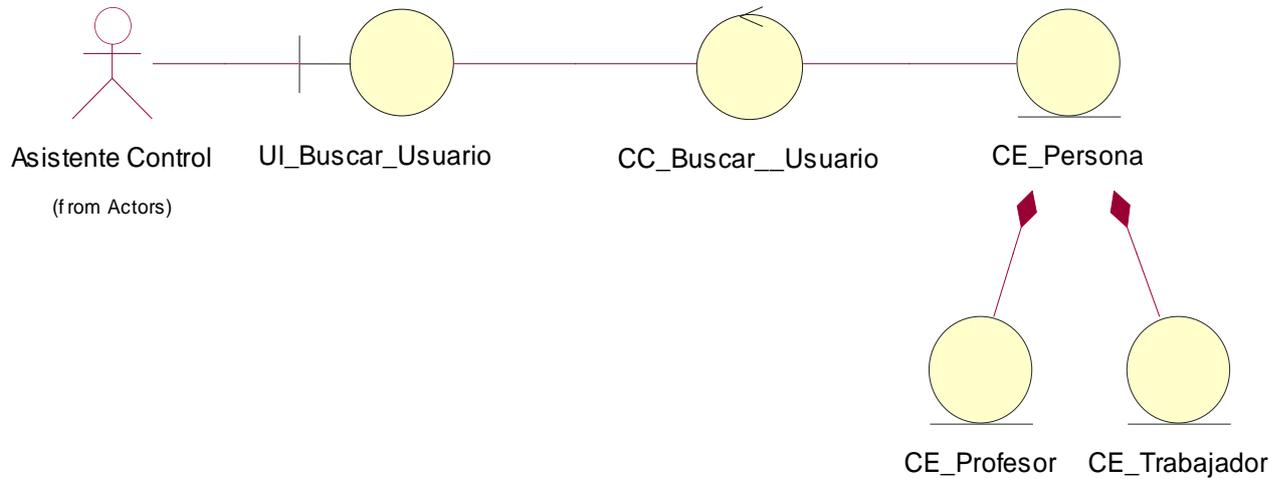
Una de las principales salidas o artefactos del análisis es el diagrama de clases de análisis. En este se representan las clases de análisis (clase Interfaz, clase controlador y clase entidad), sus relaciones entre sí y con los actores del sistema.



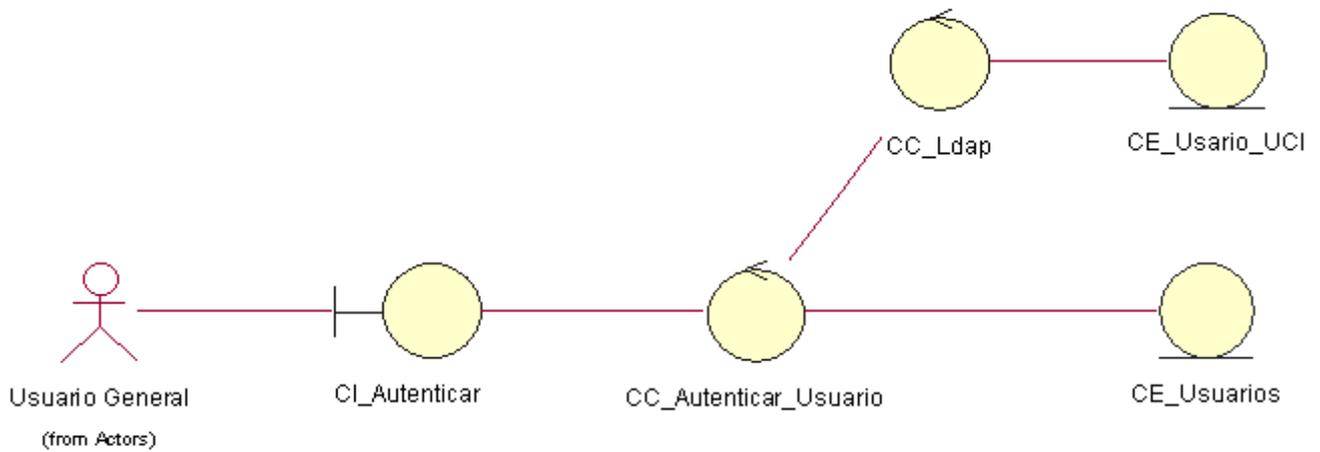
**Fig. 4.1: Diagrama de Clases del Caso de Uso Generar Reporte**



**Fig. 4.2: Diagrama de Clases del CU Gestionar Datos**



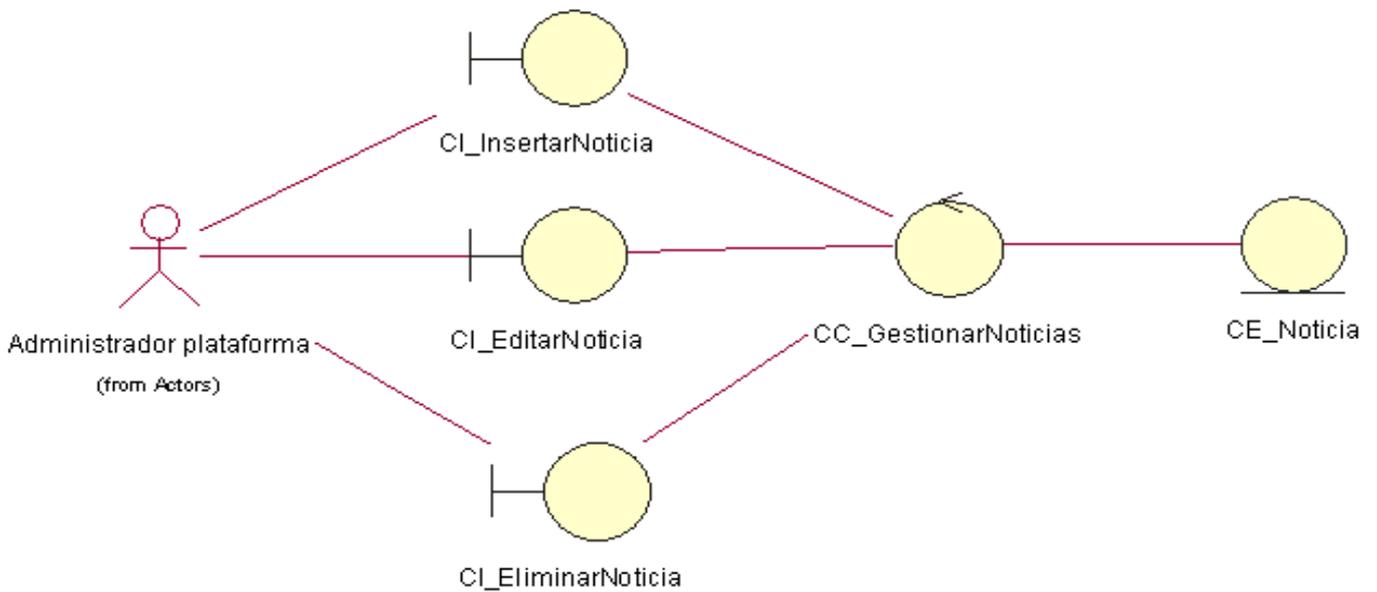
**Fig. 4.3: Diagrama de Clases del CU Buscar Usuario**



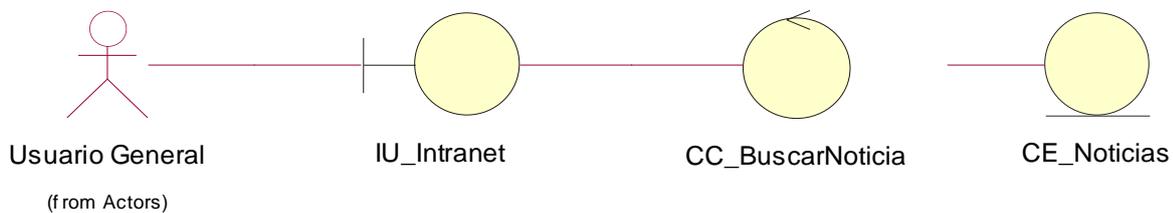
**Fig. 4.4: Diagrama de Clases del CU Autenticar**



**Fig. 4.5: Diagrama de Clases del CU Visualizar Noticia**



**Fig. 4.6: Diagrama de Clases del CU Gestionar Noticia**



**Fig. 4.7: Diagrama de Clases del CU Buscar Noticia**

## 4.2 Diseño de la intranet y el subsistema RRHH

Con el diseño se logra la comprensión de los aspectos relacionados con los requisitos no funcionales y restricciones relacionadas con los lenguajes de programación, componentes reutilizables, Sistemas Operativos, tecnologías de distribución y concurrencia y tecnologías de interfaz de usuario y se crea una entrada apropiada y un punto de partida para actividades de implementación, capturando los requisitos o subsistemas individuales, interfaces y clases.

Uno de las principales salidas del diseño lo constituyen el diagrama de clases del diseño, el modelo de datos y el diagrama de despliegue.

### 4.2.1 Diagramas de clases del diseño

Los diagramas de clases son los más utilizados en el modelado de sistemas orientados a objetos. Un diagrama de clases muestra un conjunto de clases, interfaces y colaboraciones, así como sus relaciones. Los diagramas de clases se utilizan para modelar la vista de diseño estática de un sistema, esto incluye modelar el vocabulario del sistema, modelar las colaboraciones o modelar esquemas. Los diagramas de clases también son la base para un par de diagramas relacionados: los diagramas de componentes y los diagramas de despliegue. Los diagramas de clases son importantes no sólo para visualizar, especificar y documentar modelos estructurales, sino también para construir sistemas ejecutables, aplicando ingeniería directa e inversa.[(40)]

El diagrama de clases del diseño para web, difiere un poco del resto de las aplicaciones que se construyen, puesto que en ellas son más importantes la modelación de la lógica y estado del negocio que los detalles de presentación. Para obtener un nivel correcto de abstracción y detalle que permita obtener un resultado final, es mejor modelar los artefactos del sistema, es decir: modelar las páginas, los enlaces entre estas, todo el código que irán generando las páginas, así como el contenido dinámico de estas, una vez que estén en el navegador del cliente; estos son los artefactos que se necesitan modelar para que el desarrollador los implemente luego y obtener así el producto final.

Para la confección de los diagramas de clases del diseño se tuvo en cuenta el framework Symfony y el patrón arquitectónico MVC, se dividió el modelo en cinco paquetes: el Modelo, la Vista, el Controlador, los componentes de Symfony y Propel, que es el componente que se encarga de gestionar el modelo en Symfony este contiene. El paquete Modelo contiene las clases que son utilizadas en cada caso uso del sistema. En la Vista se encuentran las páginas servidoras, las páginas clientes, el Layout y los formularios, en el caso de que los haya. En el Controlador se encuentra el controlador frontal que actúa como una página servidora, además de las clases actions que contiene todas las acciones que se implementan en el caso de uso. El paquete Symfony a su vez contiene otros paquetes que representan los componentes del framework, como por ejemplo: propel, user, controller, view.

Entre los componentes de estos cuatro paquetes se establecen diversas relaciones como <<use>>, <<submit>>, <<build>>, <<update>>.

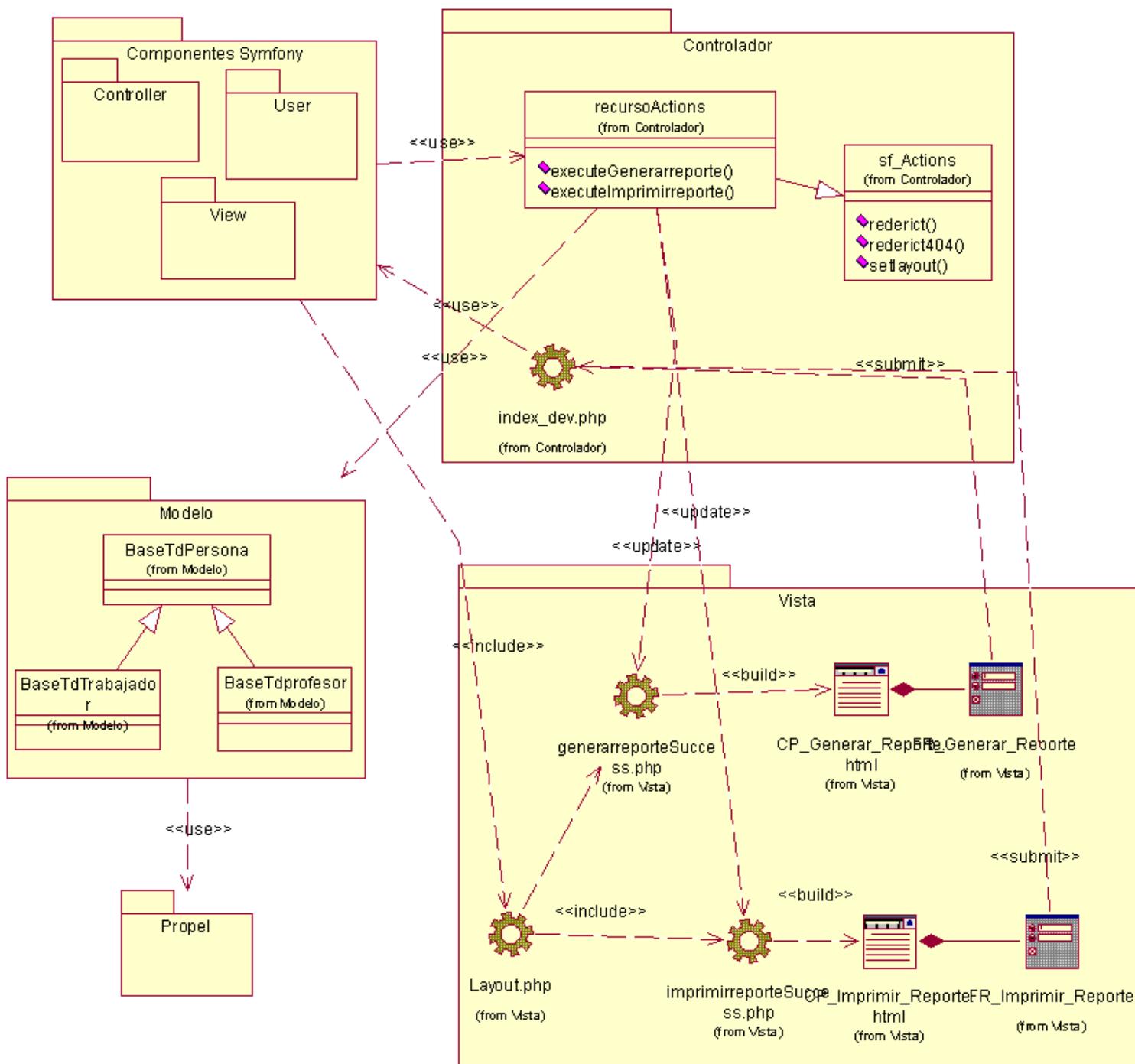


Fig 4.8: Diagrama de Clases del Diseño del CU Generar Reporte

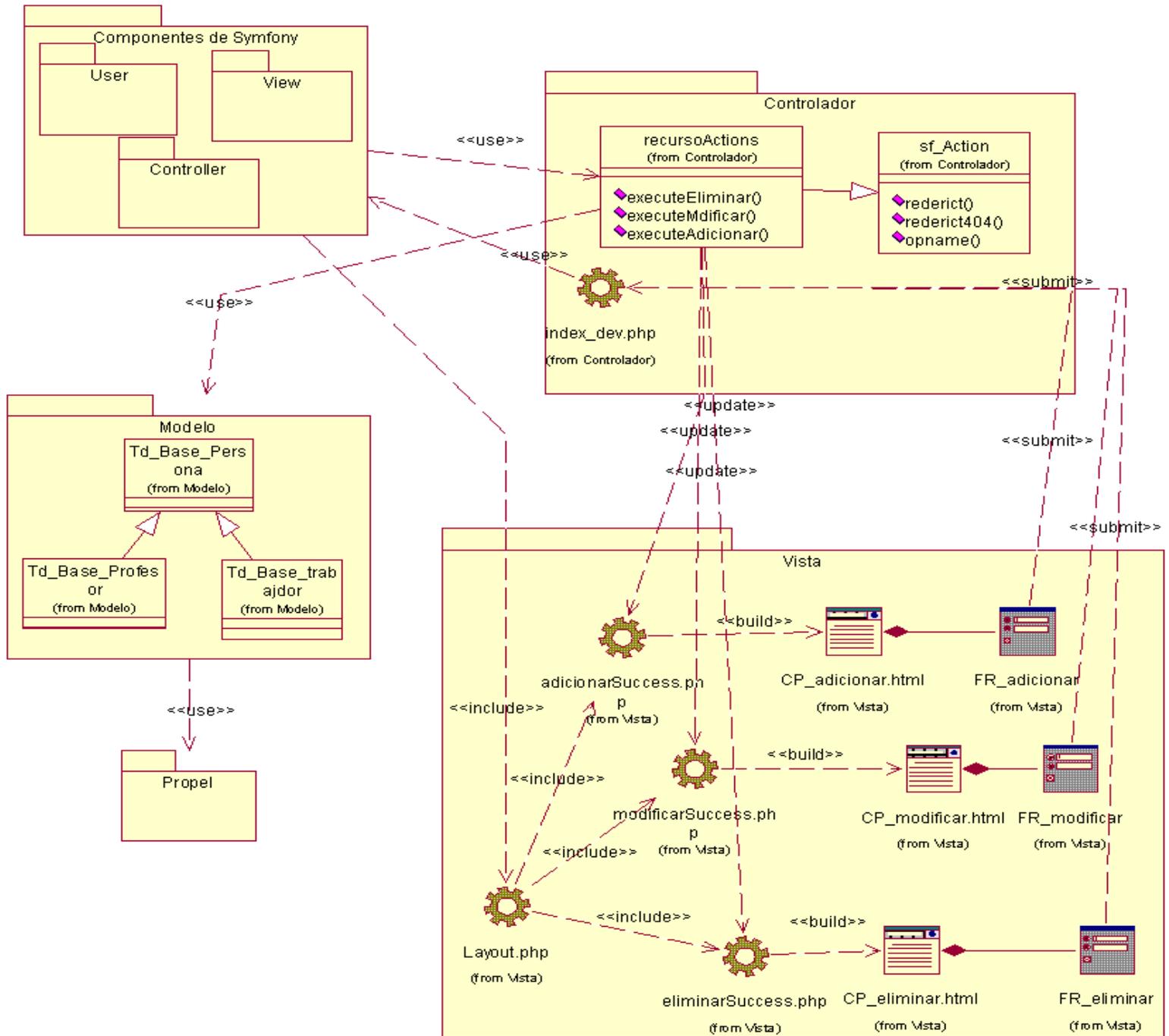


Fig 4.9: Diagrama de Clases del Diseño del CU Gestionar Datos

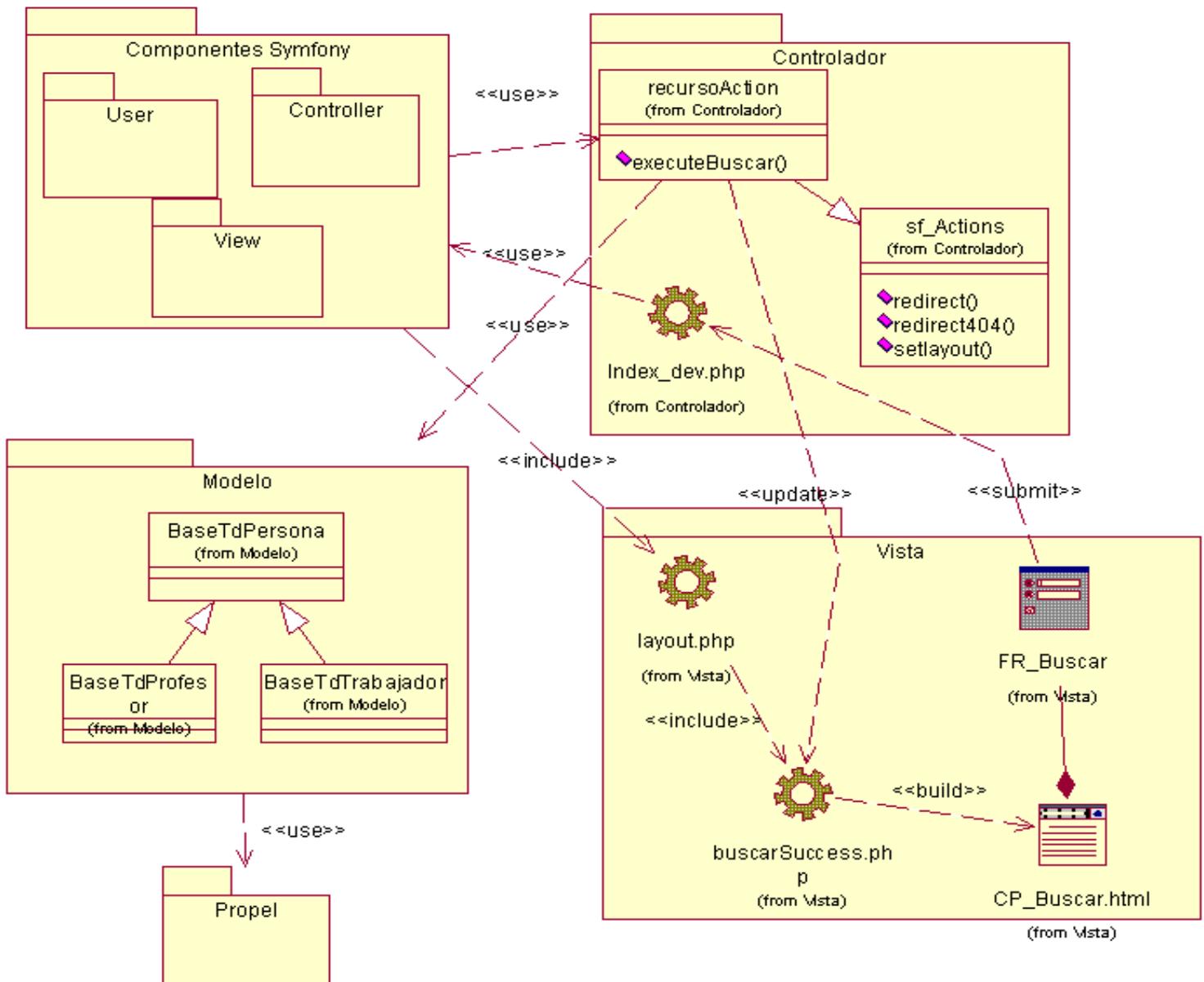


Fig 4.10: Diagrama de Clases del Diseño del CU Buscar Usuario

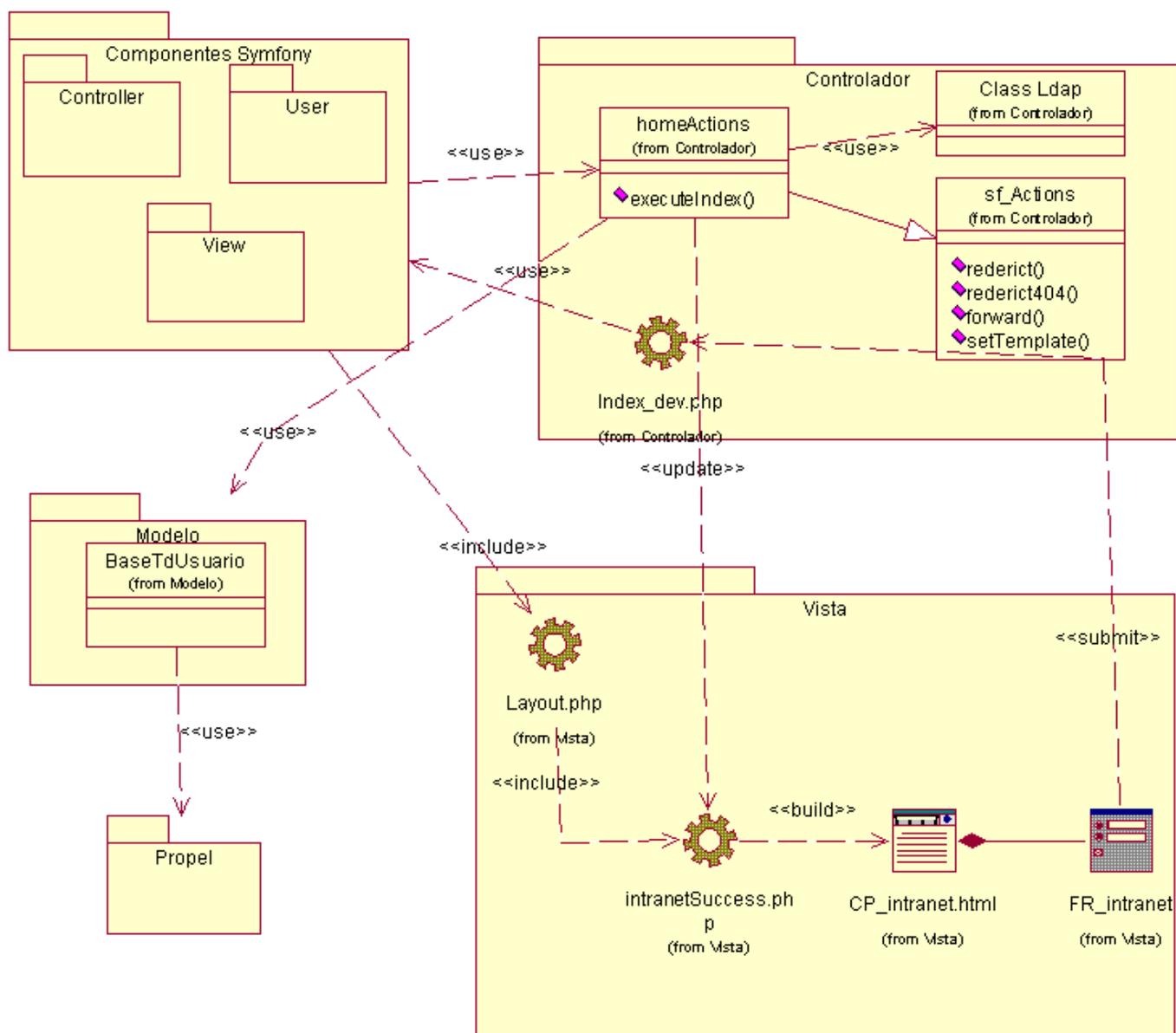


Fig 4.11: Diagrama de Clases del Diseño del CU Autenticar Usuario

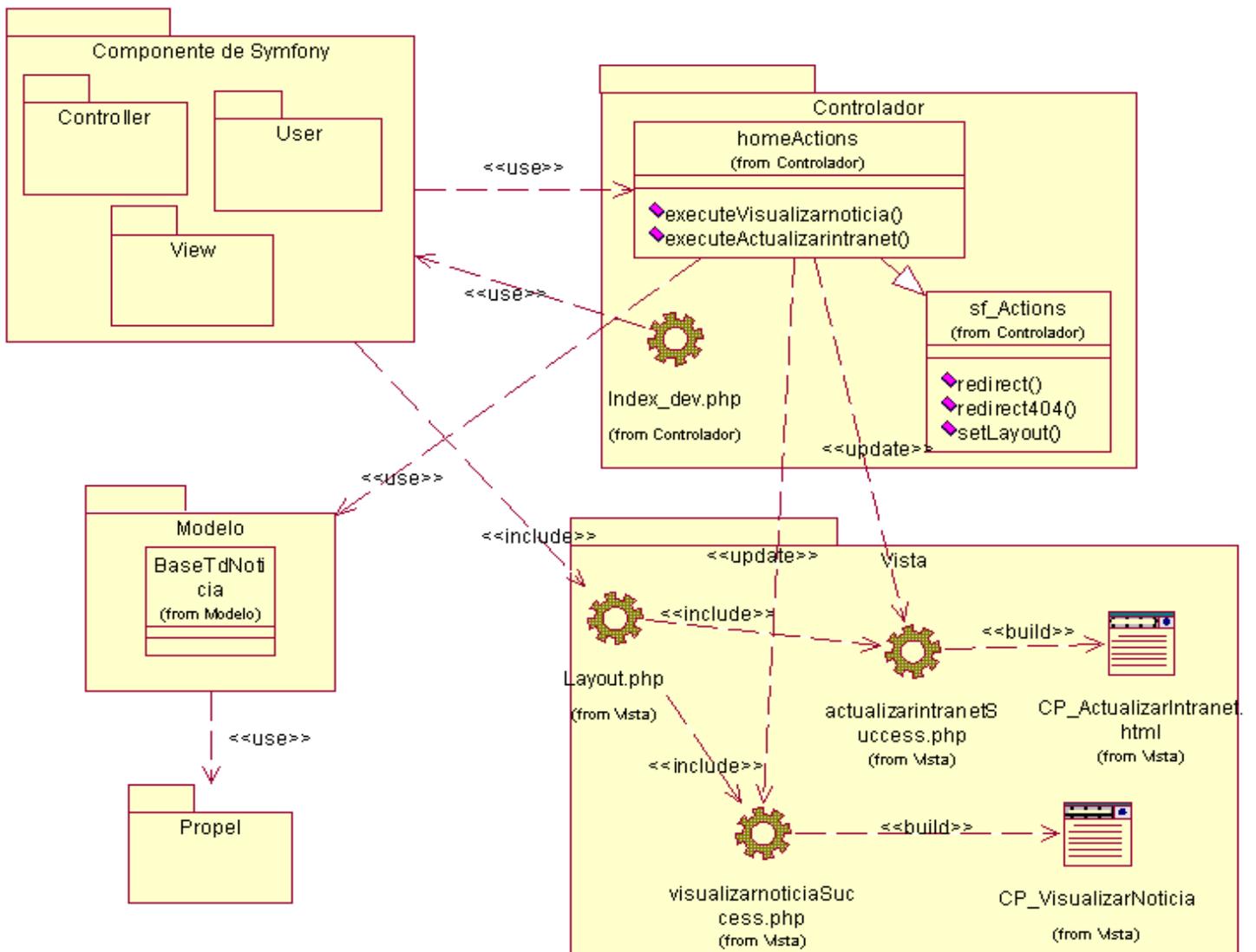


Fig 4.12: Diagrama de Clases del Diseño del CU Visualizar Noticia

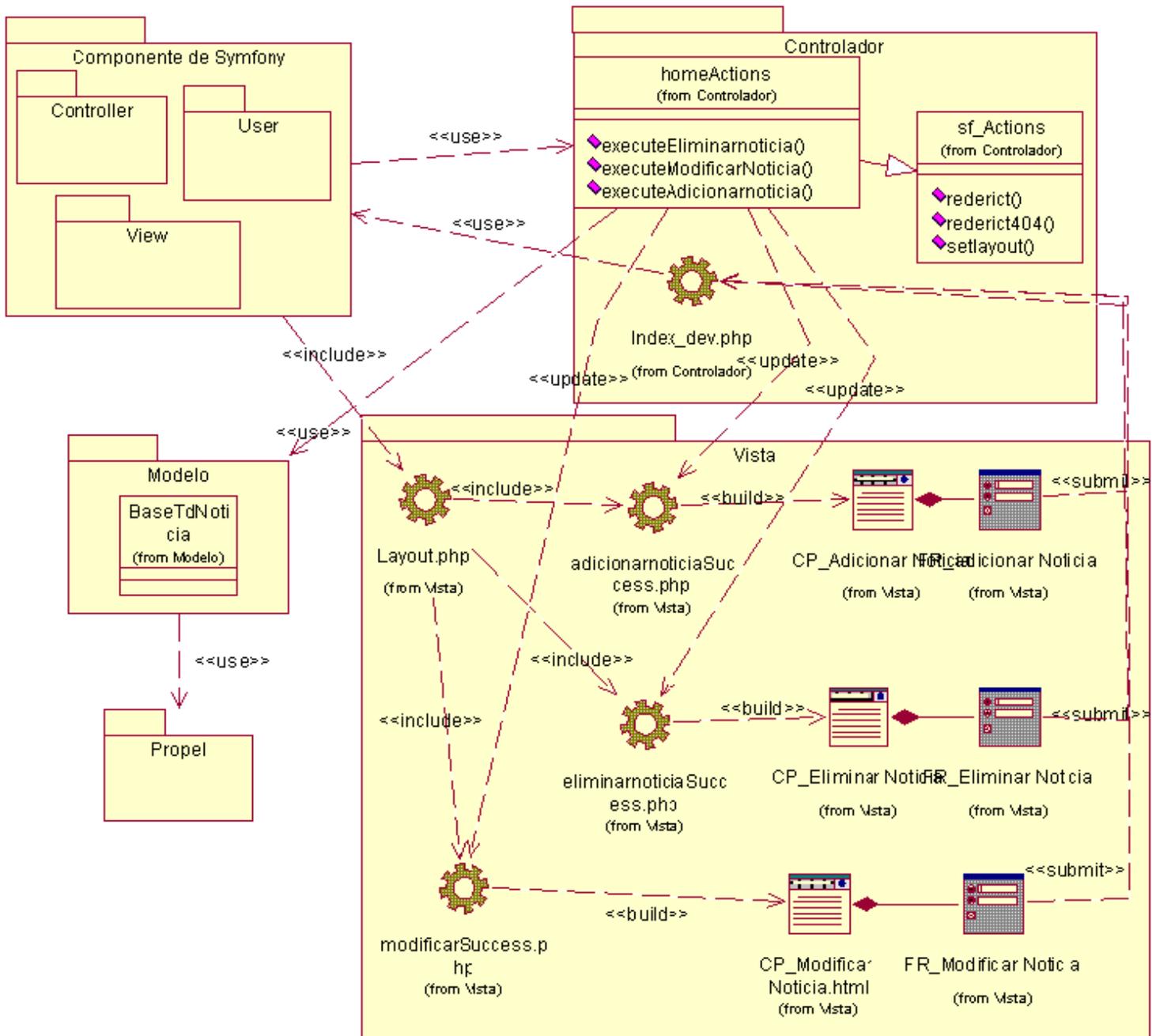


Fig 4.13: Diagrama de Clases del Diseño del CU Gestionar Noticias

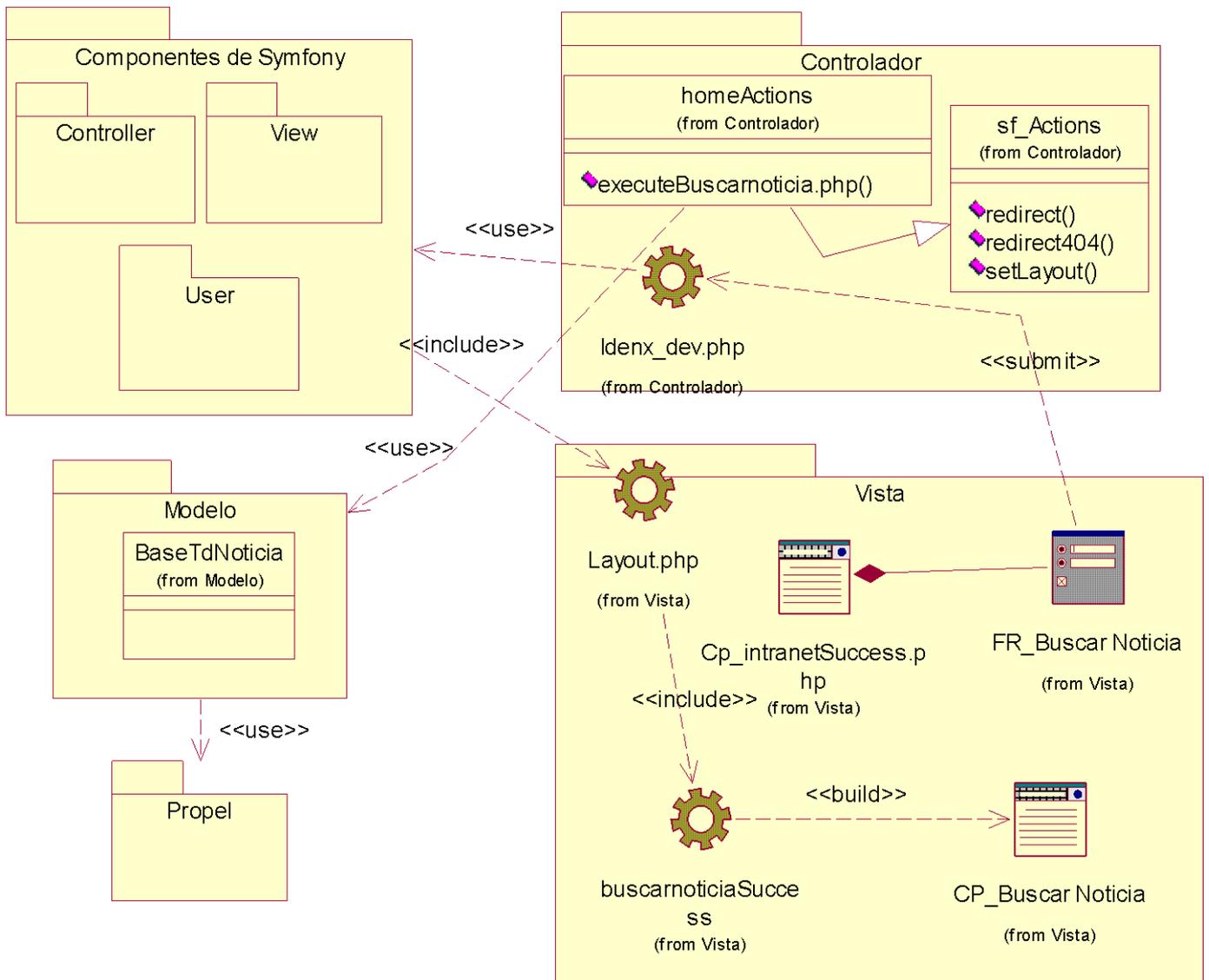


Fig 4.14: Diagrama de Clases del Diseño del CUS Buscar Noticia

### 4.3 Principios de diseño

#### 4.3.1 Estándares de la interfaz de la intranet y el subsistema RRHH

El sistema sigue la línea general de un diseño web. Se le presenta al usuario una página principal que va ser la intranet con una banner que contiene el logotipo y nombre de la aplicación. En la parte derecha inferior del banner aparece un menú que enlaza la intranet con los diferentes módulos del sistema. Esta página principal contiene noticias de la facultad y de cada área en específico. En la parte izquierda de la página se le permite al usuario autenticarse para poder acceder a los módulos que este tenga acceso en dependencia de los permisos que se le hayan dado en el sistema. Los colores predominantes de la interfaz son el azul y el gris.

En las páginas del subsistema de Recursos Humanos se mantiene el mismo banner y los colores utilizados en la intranet, en la parte izquierda se encuentran las diferentes funcionalidades que el subsistema ofrece a los usuarios. En el centro aparecen formularios según la funcionalidad que se esté ejecutando.

#### 4.3.2 Tratamiento de excepciones y seguridad del Sistema

El tratamiento de los errores es un paso fundamental para el buen funcionamiento de un sistema, garantizando la armonía y facilidad de uso de la aplicación. Para modificar los registros en la base de datos, se establecerán mecanismos que visualizarán la información para evitar los errores que puedan ocurrir y que sea válida toda la información mediante funciones JavaScript y funciones definidas por el framework utilizado. En el caso de todos los formularios de entrada de datos se validará para que en caso de haber errores, se muestren mensajes indicando los errores cometidos.

La seguridad del sistema reside principalmente en que cada usuario va a tener permisos en dependencia de las responsabilidades que tenga. En la página principal de la intranet al autenticarse un usuario, en dependencia de si es estudiante, trabajador o profesor, podrá acceder a los módulos del sistema que cada uno de estos tenga permitido acceder. En cada subsistema a los usuarios se les dan permisos para ejecutar acciones según el cargo y las responsabilidades que tenga.

#### 4.3.3 Patrones para asignar responsabilidades

En los **patrones GRASP** se reflejan algunos de los principios, que se aplican al preparar los diagramas de interacción cuando se asignan responsabilidades. GRASP es un acrónimo que significa General Responsibility Assignment Software Patterns. El nombre se eligió para indicar la importancia de captar

estos principios, si se quiere diseñar eficazmente un software orientado a objetos. Los patrones GRASP[(16)] que Symfony utiliza son:

**Experto:** es un patrón que se utiliza más que cualquiera a la hora de asignar responsabilidades. Expresan que los objetos hacen cosas de acuerdo a la información con que cuentan. Esto trae como beneficio que se conserve el encapsulamiento, además de favorecer el bajo acoplamiento.

Este es uno de los más utilizados, puesto que Propel es la librería externa que utiliza Symfony para realizar su capa de abstracción en el modelo, encapsula toda la lógica de los datos y son generadas las clases con todas las funcionalidades comunes de las entidades.

**Creador:** el propósito fundamental de este patrón es encontrar un creador que se debe conectar con el objeto producido en cualquier evento. Al escogerlo como controlador, se da soporte al bajo acoplamiento.

En la clase Actions se encuentran todas las acciones definidas. En las acciones se crean los objetos de las clases que representan las entidades, evidenciando de este modo que la clase Actions es "creador" de dichas entidades.

**Alta Cohesión:** Symfony permite asignar responsabilidades con una alta cohesión, por ejemplo la clase Actions tiene la responsabilidad de definir las acciones para las plantillas y colabora con otras para realizar diferentes operaciones, instanciar objetos y acceder a las propiedades, es decir, está formada por diferentes funcionalidades que se encuentran estrechamente relacionadas proporcionando que el software sea flexible frente a grandes cambios.

**Controlador:** todas las peticiones web son manejadas por un solo controlador frontal, que es el punto de entrada único de toda la aplicación en un entorno determinado. Cuando el controlador frontal recibe una petición, utiliza el sistema de enrutamiento para asociar el nombre de una acción y el nombre de un módulo con la URL entrada por el usuario.

**Bajo Acoplamiento:** la clase Actions hereda solamente de sfActions para lograr un bajo acoplamiento de clases.

Los **patrones GOF** que Symfony utiliza son:

**Decorator (Envoltorio):** añade funcionalidad a una clase, dinámicamente. El archivo `layout.php`, almacena el código HTML que es común a todas las páginas de la aplicación, para no tener que repetirlo en cada página. El contenido de la plantilla se integra en el layout, o si se mira desde el otro punto de vista, el layout decora la plantilla. Este comportamiento es una implementación del patrón de diseño Composite.

**Composite** (Objeto compuesto), permite tratar objetos compuestos como si de uno simple se tratase. Sirve para construir objetos complejos a partir de otros más simples y similares entre sí, gracias a la composición recursiva y a una estructura en forma de árbol. Esto simplifica el tratamiento de los objetos creados, ya que al poseer todos ellos una interfaz común, se tratan todos de la misma manera.

**Singleton (Instancia única):** garantiza la existencia de una única instancia para una clase y la creación de un mecanismo de acceso global a dicha instancia. En el controlador frontal hay una llamada a `sfContext::getInstance()`. En una acción, el método `getContext()`, un objeto muy útil que guarda una referencia a todos los objetos del núcleo de Symfony.

**Abstract Factory (Fábrica abstracta):** permite trabajar con objetos de distintas familias de manera que las familias no se mezclen entre sí y haciendo transparente el tipo de familia concreta que se esté usando. Cuando el framework necesita por ejemplo crear un nuevo objeto para una petición, busca en la definición de la factoría el nombre de la clase que se debe utilizar para esta tarea.

#### 4.4 Integración con servicios web

Un servicio web es un conjunto de protocolos y estándares que sirven para intercambiar datos entre aplicaciones. Distintas aplicaciones de software desarrolladas en lenguajes de programación diferentes, y ejecutadas sobre cualquier plataforma, pueden utilizar los servicios web para intercambiar datos en redes de ordenadores como Internet. La interoperabilidad se consigue mediante la adopción de estándares abiertos.

El objetivo que se persigue con la base de datos del sistema es que sea actualizada con los servicios web que existen en la universidad. De esta forma no existirá réplica de información, en el caso de que un profesor, trabajador o estudiante sea baja o nuevo ingreso de la universidad la base de datos se actualizará directamente de los servicios web que ya están implementados en algunos sistemas de la

universidad como son: el Sistema de Gestión Académica, Registro de Identidad UCI y el Sistema de Capital Humano. De cada uno de los sistemas anteriormente mencionados se van obtener la mayoría de los datos de los estudiantes, profesores y trabajadores de la facultad 9.

Si no existe comunicación con estos servicios web la persona encargada del control de la base de datos tendrá que estar constantemente pendiente de las bajas y nuevos ingresos para actualizar manualmente la misma.

### **4.5 Diseño de la Base de Datos de la Plataforma**

Una de las tareas trazadas al inicio del desarrollo de la tesis fue diseñar una base de datos centralizada para los seis subsistemas y la Intranet. La base de datos está compuesta por 77 tablas, debido a este número tan alto solamente se representan las relaciones entre las tablas utilizadas en el Subsistema RRHH. En el anexo se muestran los modelos de datos de los restantes subsistemas.

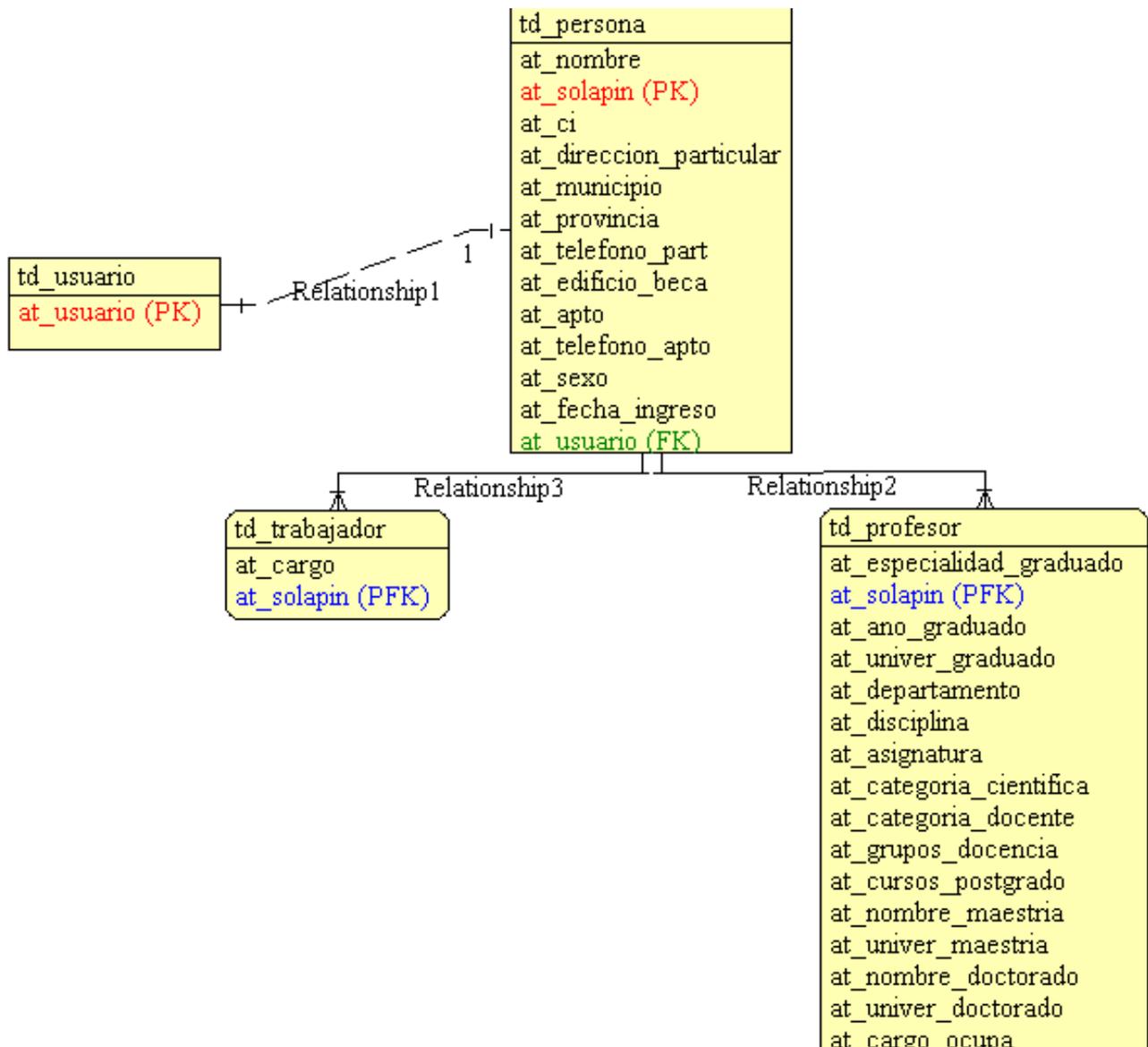
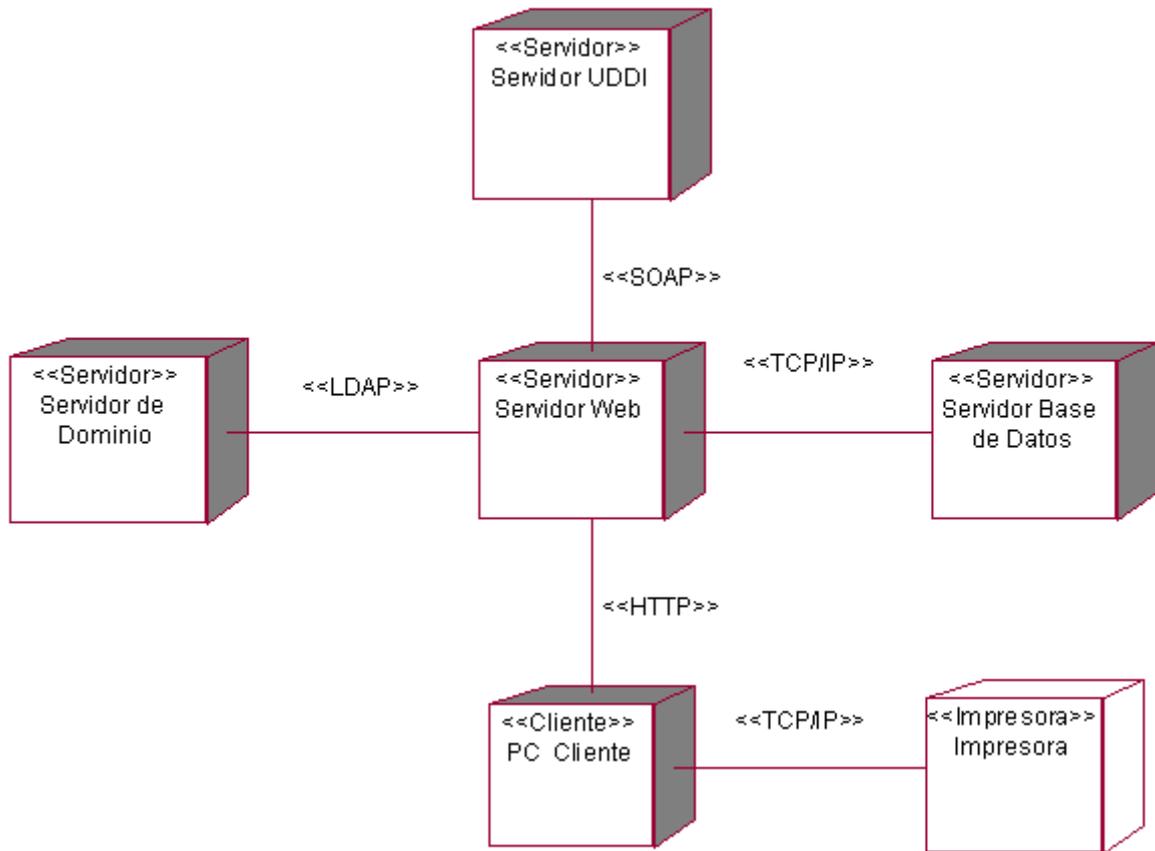


Fig 4.15: Modelo de datos del Subsistema RRHH

#### 4.6 Modelo de Despliegue del subsistema RRHH

Un Diagrama de Despliegue es un diagrama que muestra la configuración de los nodos que participan en la ejecución. Se modelan la topología del hardware sobre el que se ejecuta el sistema y la distribución física del sistema.

Para la puesta en práctica de la aplicación se necesitará principalmente tres procesadores: uno que servirá como servidor de la aplicación, el cual se encontrará conectado directamente al servidor de base de datos mediante el protocolo de comunicación TCP/IP y la PC cliente mediante el protocolo



HTTP. Las PCs clientes utilizarán una impresora siendo esta un recurso compartido conectada a la red utilizando un Print Server mediante el protocolo TCP/IP.

**Fig.4.16: Diagrama de Despliegue**

#### 4.7 Modelo de Implementación del subsistema RRHH

El Modelo de Implementación es comprendido por un conjunto de componentes y subsistemas que constituyen la composición física de la implementación del sistema. Entre los componentes se encuentran datos, archivos, ejecutables, código fuente y los directorios. Fundamentalmente, se describe la relación que existe desde los paquetes y clases del modelo de diseño a subsistemas y componentes físicos.

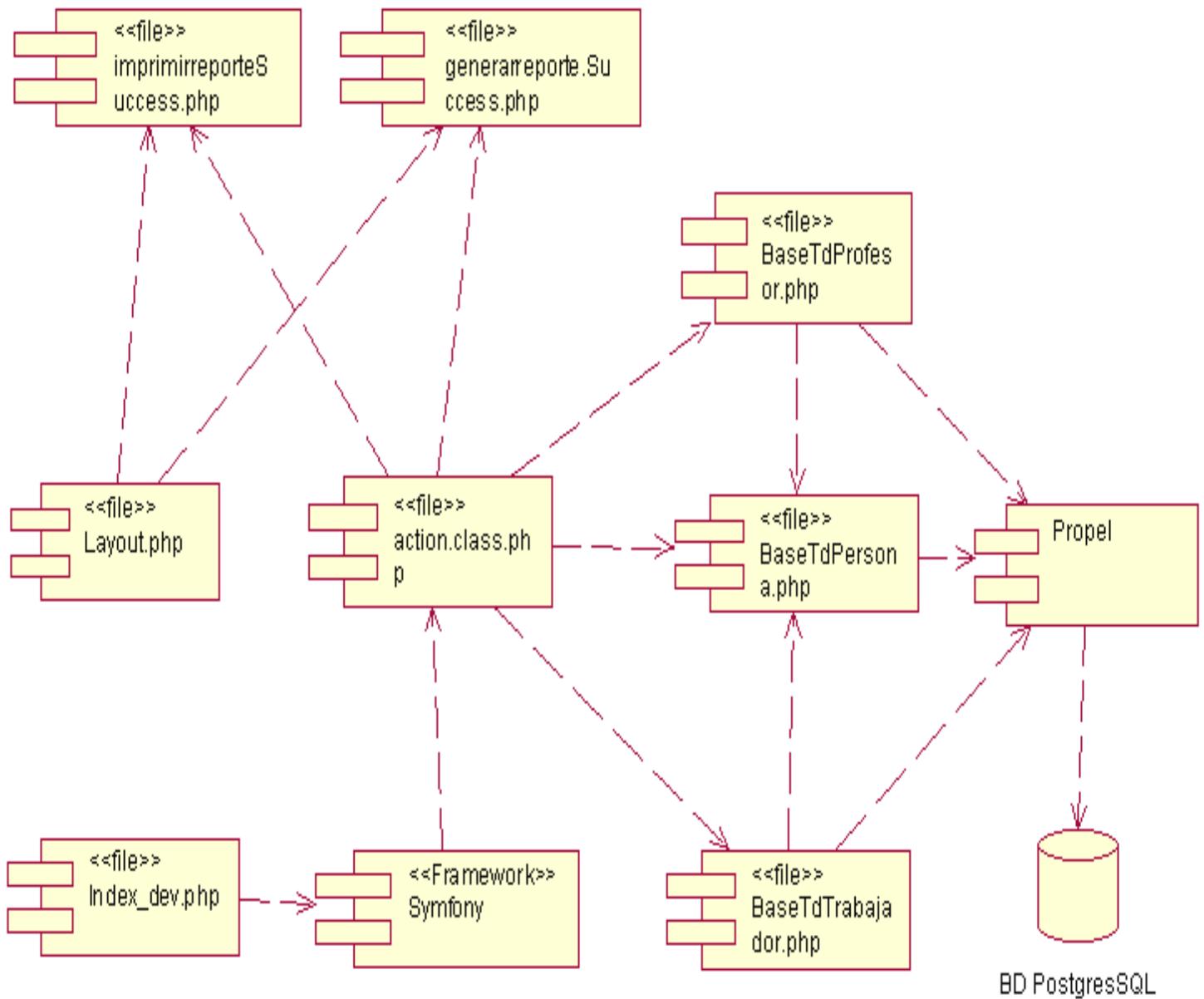


Fig 4.17: Diagrama de Componentes del CU Generar Reporte.

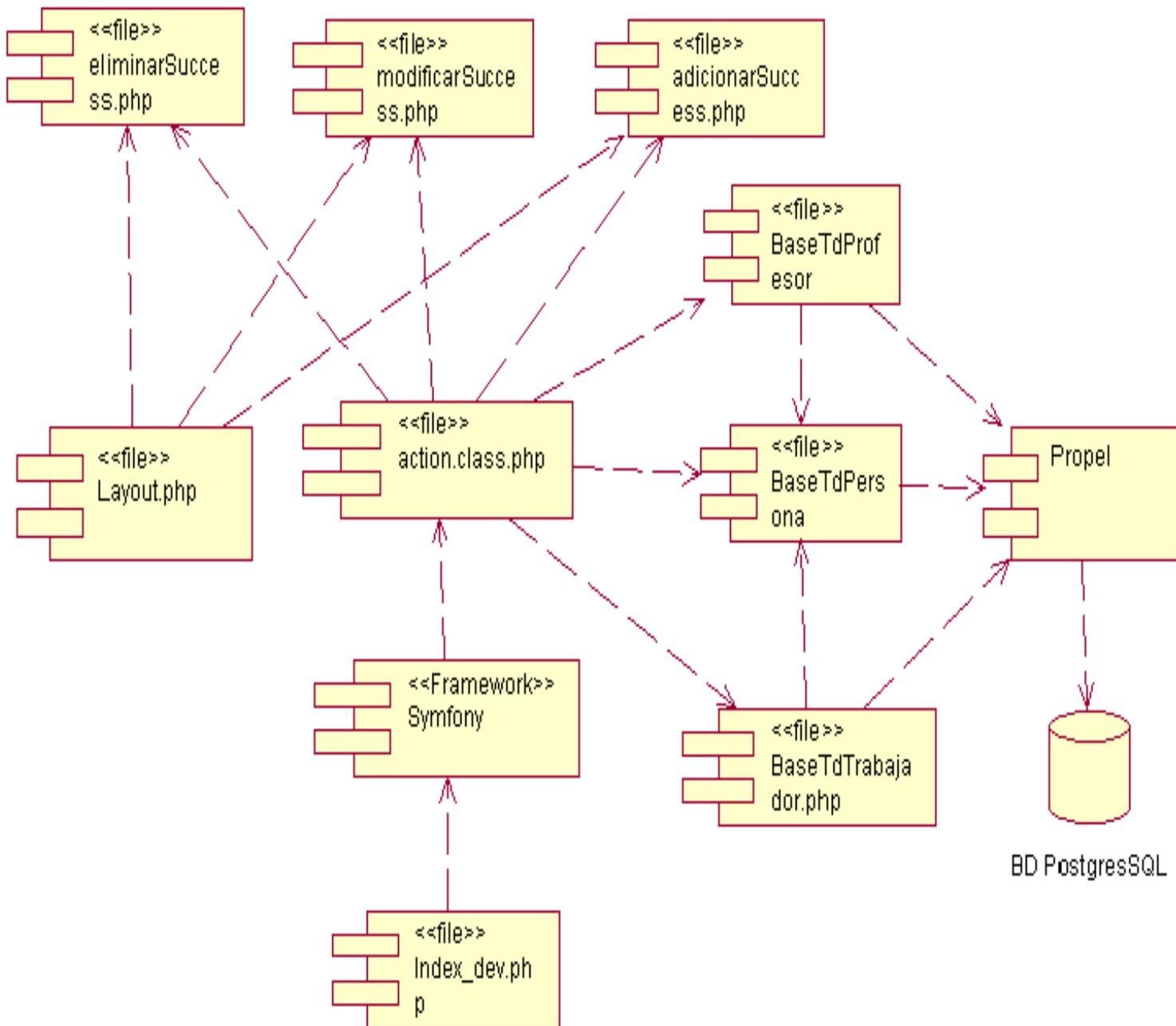


Fig 4.18: Diagrama de Componentes del CU Gestionar Datos.

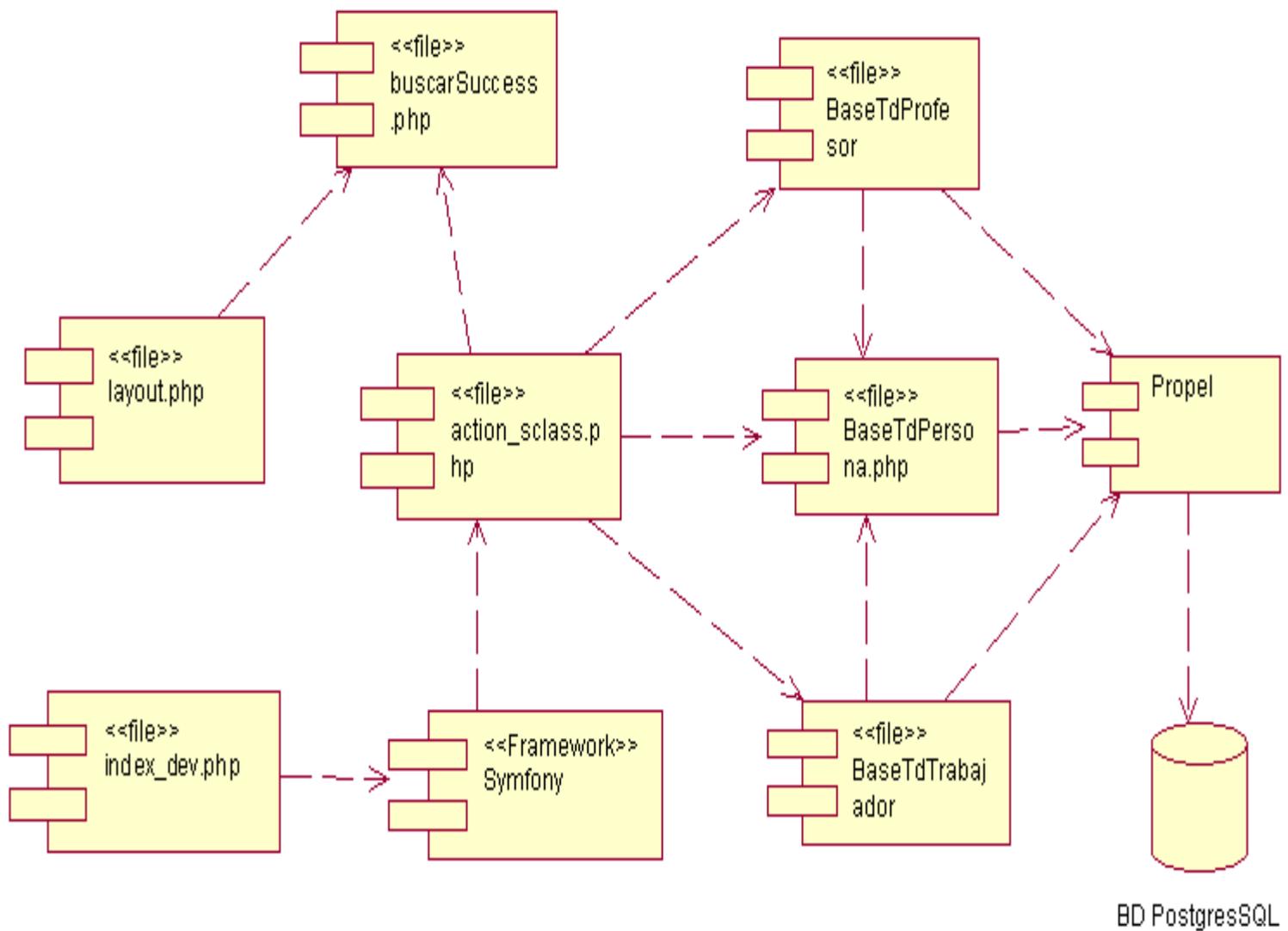


Fig 4.19: Diagrama de Componentes del CU Buscar Usuario.

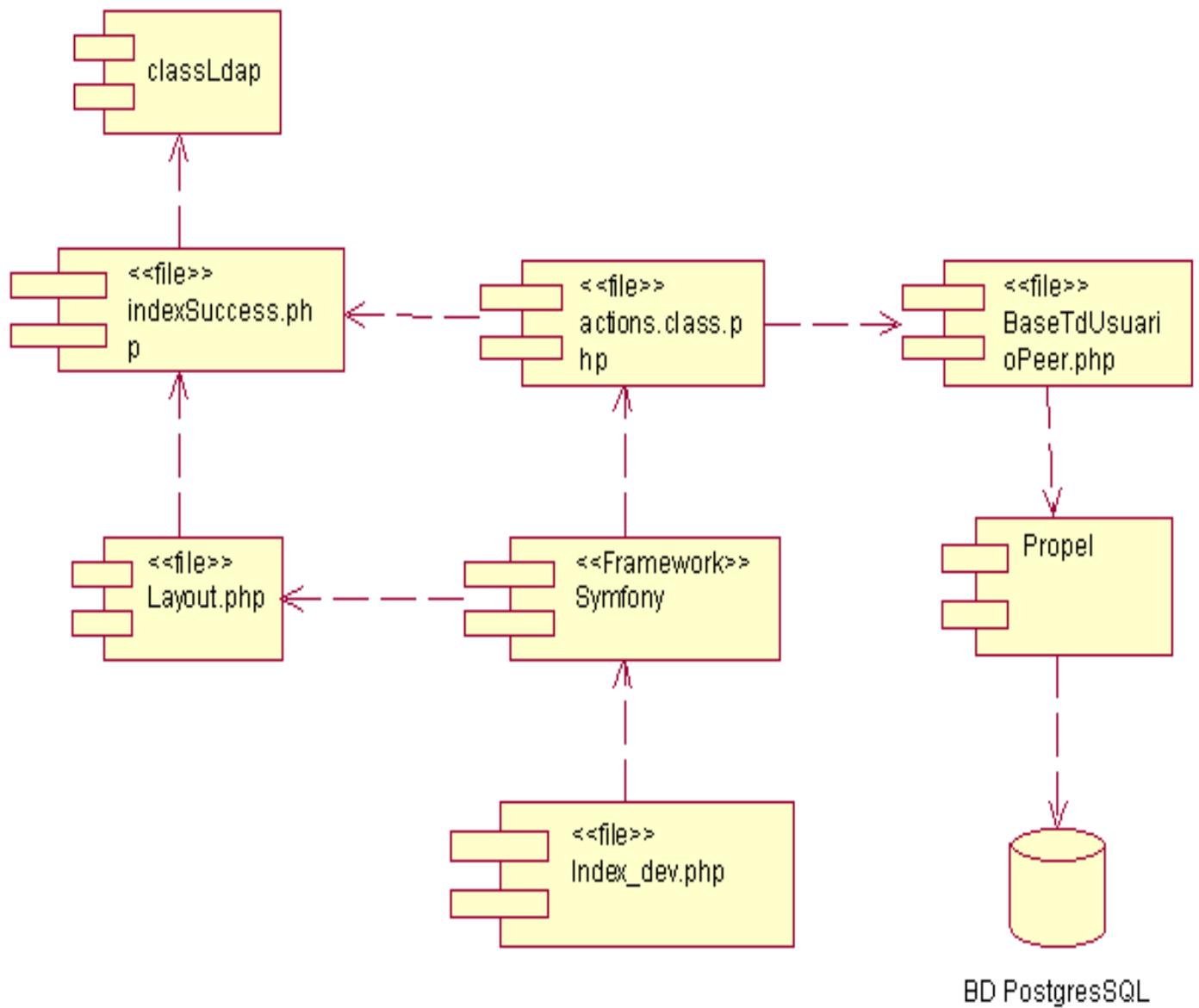


Fig 4.20: Diagrama de Componentes del CU Autenticar Usuario.

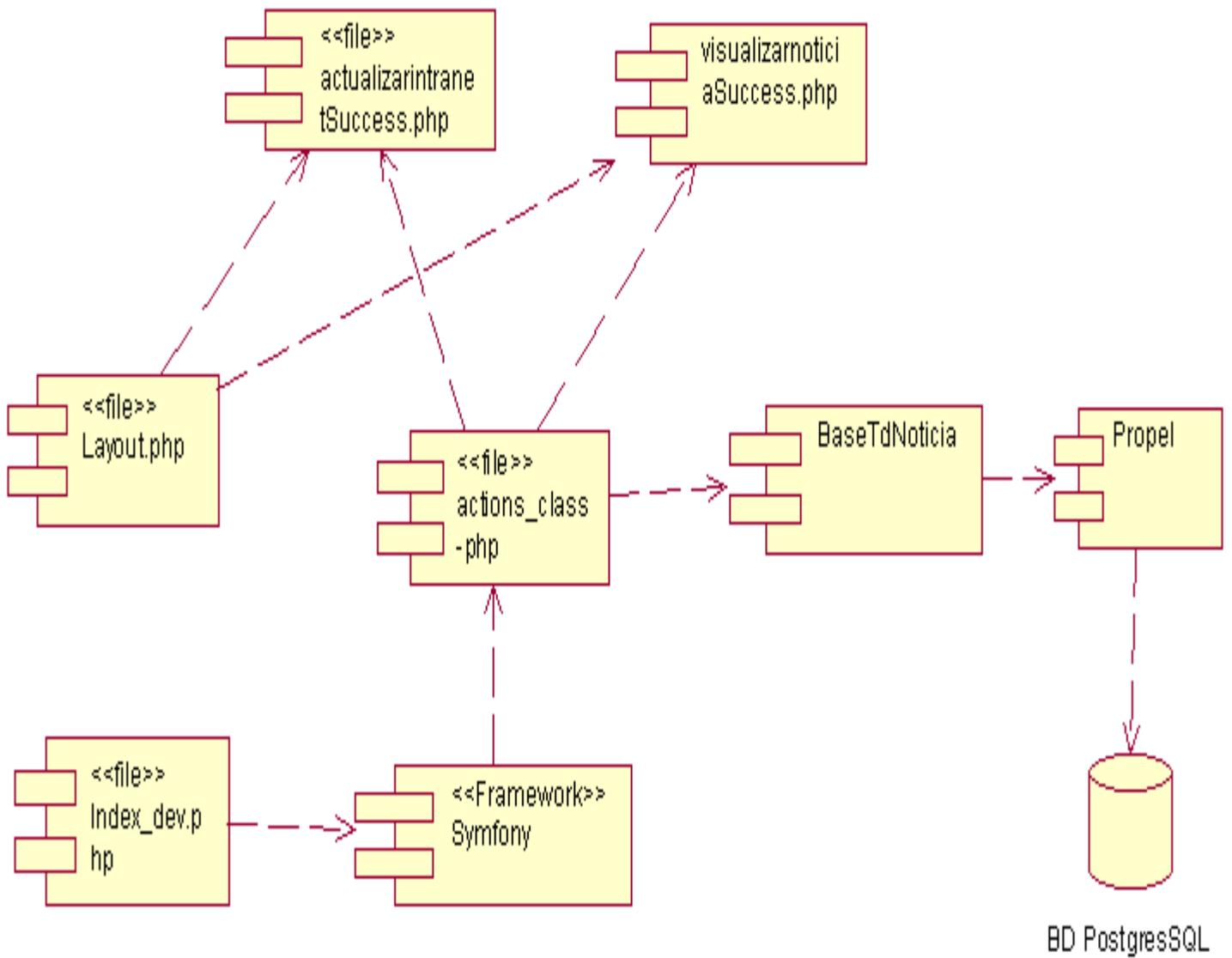


Fig 4.21: Diagrama de Componentes del CU Visualizar Noticia.

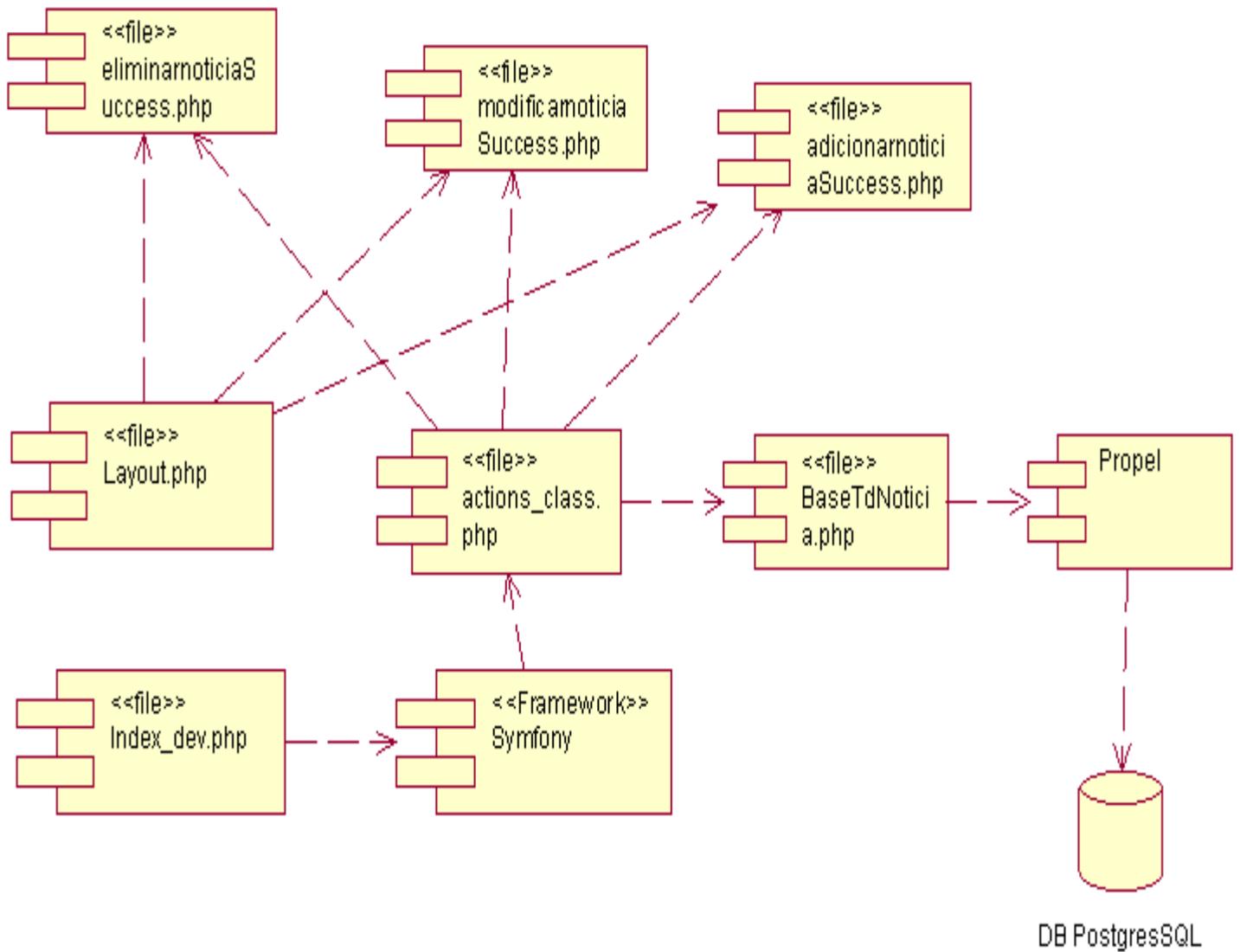


Fig 4.22: Diagrama de Componentes del CU Gestionar Noticias.

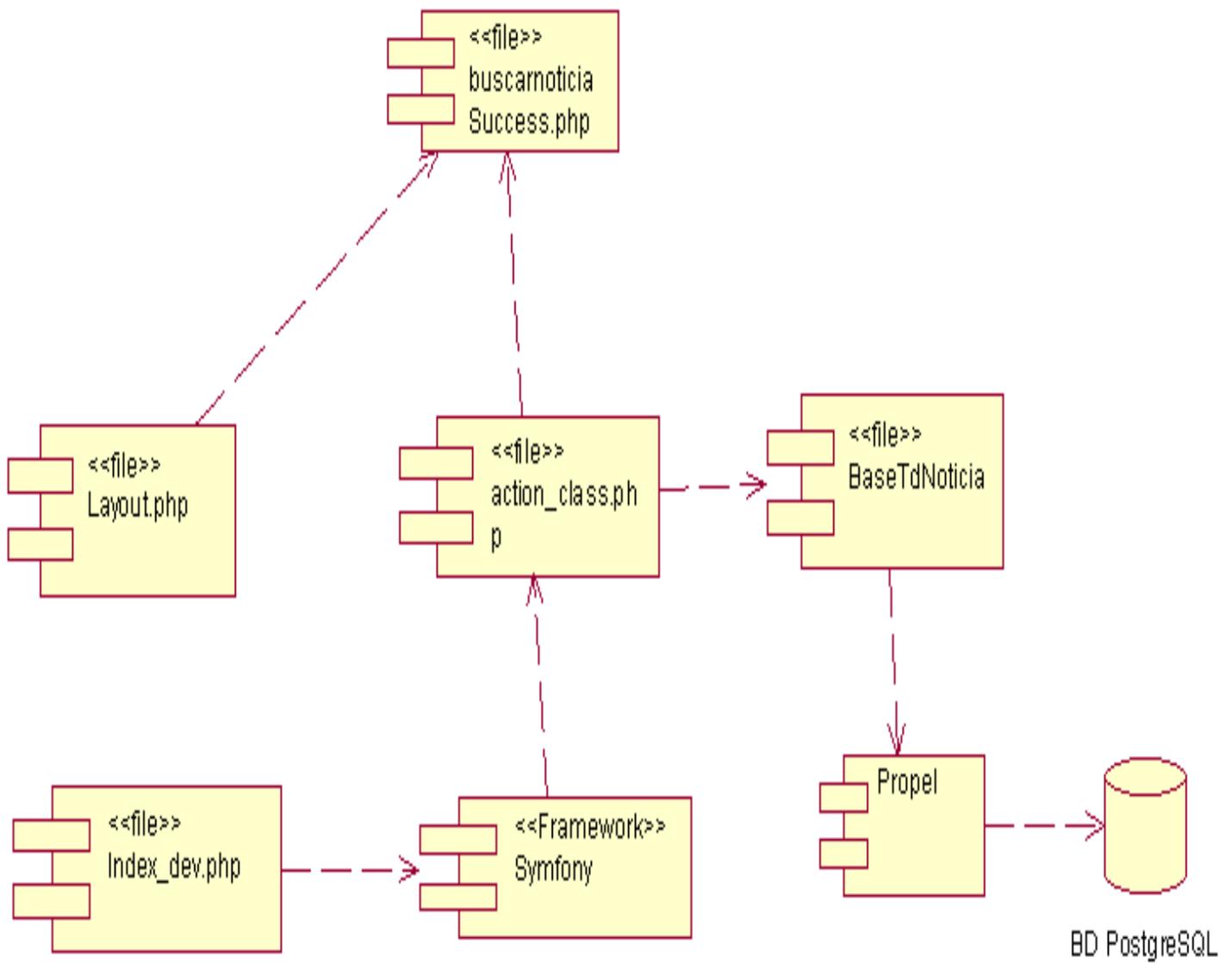


Fig 4.23: Diagrama de Componentes del CU Buscar Noticias.

### **4.8 Conclusiones Parciales**

En este capítulo se modeló detalladamente la solución propuesta a través de los modelo de análisis, diseño y los artefactos necesarios de implementación para posibilitar una mayor comprensión. Se representó mediante diagramas de clases utilizando estereotipos web y componentes de Symfony la lógica del negocio del sistema, alcanzando de esta forma un alto grado de especificidad en la realización de los casos de uso, para la asignación de responsabilidades se utilizaron patrones de diseño expuestos en este capítulo. Se mostró el modelo de datos del subsistema de Control de los Recursos Humanos así como las clases persistentes, fue realizada la modelación de los nodos en los que será distribuida la aplicación especificando para cada uno de éstos el protocolo de comunicación.

## CONCLUSIONES

A modo de conclusión se puede decir que se han cumplido exitosamente los objetivos trazados, modelándose una plataforma a la altura de lo que se quería y se necesitaba:

- Se cumplieron los requisitos solicitados por el cliente.
- La aplicación apoyada en su interfaz web, permite la rapidez en el acceso y en las búsquedas de información por parte del personal disminuyendo así su carga laboral.
- El sistema resultante está provisto de un ambiente cómodo, fácil de entender, que cumple los estándares del diseño y utiliza técnicas modernas de programación orientada a objetos.
- Se logró la centralización en una base de datos de toda la información referente a los procesos que se desarrollan en la facultad 9.
- Se utilizaron los servicios web que brinda la UCI para mantener actualizada la base de datos.
- Se logró una seguridad y protección de los datos consecuente con el nivel de seguridad requerido.
- Después de un estudio previo y mediante esta puesta en práctica, se ha demostrado la eficacia de los lenguajes y tecnologías utilizadas para el desarrollo del sistema.

## RECOMENDACIONES

- Poner en práctica la utilización de la base de datos, la Plataforma y la Intranet de la facultad 9 en las otras facultades de la universidad.
- Que se mejore el diseño de la presentación del sistema.
- Que se perfeccionen y amplíen los servicios que brinda el sistema.
- Que se refine y actualice la información de la base de datos.

## REFERENCIAS BIBLIOGRÁFICAS

1. WAYNE R, M. Y. N., ROBERT M. *Administración de Recursos Humanos*. México Editorial PEARSON, 1997. 345 p. ISBN 970-26-0641-1.
2. BELAMARIC, R. A. *Capital Humano. Concepto e instrumentación*. publicado el: 16/3/2008 de última actualización: 16/3/2008. [Consultado el: 06/01/2008]. Disponible en: [http://www.nodo50.org/cubasiqloXXI/pensamiento/alhama\\_311204.htm](http://www.nodo50.org/cubasiqloXXI/pensamiento/alhama_311204.htm).
3. KIOSMY ALMENARES HERRERA, R. D. L. B. *Diseño e Implementación del proceso de inscripción del Módulo Mercantil en las Oficinas Registrales de la República Bolivariana de Venezuela*. 2007.
4. *Plataforma de desarrollo*. publicado el: 26/3/2008 de 2005, última actualización: 26/3/2008. Disponible en: [http://es.wikipedia.org/wiki/Plataforma\\_de\\_desarrollo](http://es.wikipedia.org/wiki/Plataforma_de_desarrollo).
5. INTEL. Disponible en: <http://www.intel.com/cd/corporate/techtrends/emea/spa/platforms/325004.htm>.
6. GIACOMANTONIO, M. *Dove vanno le piattaforme di e-Learning*. 2004 Disponible en: [http://www.wbt.it/index.php?risorsa=piattaforme\\_elearning](http://www.wbt.it/index.php?risorsa=piattaforme_elearning)
7. BENEDETTO, I. "Dalla valutazione dell'apprendimento alla valutazione dell'ambiente di apprendimento", en *Form@re Erickson* 2001, Disponible en: [http://www.formare.erickson.it/archivio/settembre/2\\_benedetto.html](http://www.formare.erickson.it/archivio/settembre/2_benedetto.html).
8. BELYK, D. F., D. . "Software evaluation criteria and terminology", en *The Center of Distance Education (Athabasca University)* publicado el: 22/3/2008 de 2002, última actualización: 22/3/2008. Disponible en: <http://cde.athabascau.ca/softeval/reports/R070203.pdf>.
9. ROMÁN MENDOZA, E. C., P. AND PÉREZ, P. . *Web-based Instructional Environments: Tools and Techniques for effective Second Language Acquisition*. Servicio de Publicaciones de la Universidad de Murcia,

10. AL, Q. La importancia de la intranet como medio de comunicación. nº Disponible en: <http://www.miespacio.org/cont/trin/intrane.htm>.
11. PÉREZ, J. A. U. *La información en Internet*. Mexico: 2002. vol. Vol.11,
12. H., P. P. Taller en Tecnología de Redes Internet para América Latina y el Caribe, INET'98, programa "Bibliotecarios en Tecnologías Internet", promovido por la Internet Society, ENRED y Mercosur. 1998 nº [Consultado el: 13/01/2008]. Disponible en: <http://biblioteca.udea.edu.co/~hlopera/intranet.html>.
13. PUJOL, M. E. F. I. *Internet, Intranets, Extranets, ¿Son importantes en la empresa periodística?* publicado el: 22/42008 de última actualización: 22/42008. Disponible en: <http://www.ucm.es/info/multidoc/multidoc/revista/cuad6-7/eulalia.htm>.
14. LLOP, J. M. T. *Ataque de denegación de servicio*. 2004.
15. BELLIDO, A. *Estándares web*. Disponible en: <http://www.maestrosdelweb.com/editorial/estandaresweb/>.
16. *Patrones GRASP*. publicado el: 26/05/2008 de última actualización: 26/05/2008. Disponible en: <http://jorgesaavedra.wordpress.com/2006/08/17/patrones-grasp-craig-larman/>.
17. *Características de SQL Server*. publicado el: 16/04/2008 de última actualización: 16/04/2008. Disponible en: <http://www.shica19.tripod.com/sql.html>, .
18. *Article\_MySQL-PostgreSQL*. Disponible en: <http://www.phpbuilder.com/columns/tim20000705.php3?page=1> .
19. *Ventajas de PostGreSQL*. publicado el: 09/01/2008 de 2007, última actualización: 09/01/2008. [Consultado el: 09/01/2008]. Disponible en: [http://soporte.tiendalinux.com/portal/Portfolio/postgresql\\_ventajas.html](http://soporte.tiendalinux.com/portal/Portfolio/postgresql_ventajas.html).

20. *Introducción a la ASP.NET.* publicado el: 16/04/2008 de última actualización: 16/04/2008. Disponible en: [http://www.gamarod.com.ar/articulos/introduccion\\_a\\_aspnet.asp](http://www.gamarod.com.ar/articulos/introduccion_a_aspnet.asp).
21. *Las diez características de JSP.* publicado el: 16/04/2008 de última actualización: 16/04/2008. Disponible en: <http://blog.taragana.com/index.php/archive/top-10-java-features-or-what-makes-java-great/es/>.
22. MEHDI ACHOUR, F. B., ANTONY DOVGAL. 2007. *PHP. Manuales.* 2007, Disponible en: . <http://es.php.net/manual/es/>.
23. *Los mejores IDE's para PHP.* publicado el: 18/03/2008 de 2007, última actualización: 18/03/2008. Disponible en: <http://www.tufuncion.com/ide-php>.
24. 18/03/2008 2008, nº Disponible en: [http://www.phpeclipse.net/tiki-view\\_articles.php](http://www.phpeclipse.net/tiki-view_articles.php).
25. SÁNCHEZ, M. A. M. *Metodologías De Desarrollo De Software* 2004, Disponible en: <http://www.informatizate.net>
26. IVAR JACOBSON, G. B., JAMES RUMBAUGH. *El Proceso Unificado de Desarrollo de Software.* Addison Wesley ed. España: 2002. 458 p. ISBN 84-7829-036-2.
27. LARMAN, C. *UML y Patrones.* Dawn Speth White ed. Mexico: 1999. 536 p. ISBN 970-17-0261-1.
28. EARL, A. *CASE Tool Information.* Disponible en: <http://www.cs.queensu.ca/Software-Engineering/case.html>.
29. RATIONAL. 2007, Disponible en: <http://www.rational.com/uml/>.
30. *Hay que buscar el nombre en la red.* publicado el: 22/3/2008 de última actualización: 22/3/2008. Disponible en: [http://www.hospedajeydominios.com/mambo/documentacion-manual\\_apache-pagina-46.html](http://www.hospedajeydominios.com/mambo/documentacion-manual_apache-pagina-46.html).

31. GUTIÉRREZ, A. F. *Libro de Kumbia: Porque Programar debería ser más fácil*. publicado el: 03/02/08 de última actualización: 03/02/08. Disponible en: <http://bitelia.com/2007/09/22/kumbia-php-framework/>.
32. *Symfony, open-source PHP web framework*. 2008, Disponible en: <http://frameworkphp.wordpress.com/2007/08/31/symfony-open-source-php-web-framework/>.
33. FABIEN POTENCIER, F. Z. *Symfony: La guía definitiva*. 2008, Disponible en: <http://www.librosweb.es/symfony/pdf/capitulo1.pdf>
34. ---. Disponible en: [http://www.librosweb.es/symfony/capitulo2/el\\_patron\\_mvc.html](http://www.librosweb.es/symfony/capitulo2/el_patron_mvc.html).
35. *Utilice un framework MVC comprobado*. publicado el: 2/4/2008 de 2008, última actualización: 2/4/2008. Disponible en: <http://espanol.aurumsol.com/articulos/art7/art7-8.html>.
36. ACADEMIA, R. *ADOBE Principales funciones de ADOBE DreamWeaver*. 2007, Disponible en: <http://www.macromedia.com/es/software/dreamweaver/>.
37. publicado el: 18/03/2008 de 2007, última actualización: 18/03/2008. Disponible en: [http://www.freedownloadcenter.com/es/Programacion/Base\\_de\\_Datos\\_y\\_Redres/CAS\\_E\\_Studio\\_2.html](http://www.freedownloadcenter.com/es/Programacion/Base_de_Datos_y_Redres/CAS_E_Studio_2.html).
38. ALICIA MARTÍNEZ R, H. E. E., OSCAR PASTOR *El Modelo de Negocios como origen de especificaciones de requisitos de software*. Mexico: 2002.
39. MARÍA JOSÉ ORTÍZ, J. G. M., BEGOÑA MOROS, JOAQUÍN NICOLÁS. *El Modelo del Negocio como base del Modelo de Requisitos*. 2000, nº
40. JACOBSON, I. G. B., ET AL. *El proceso Unificado de Desarrollo de Software*. Eddison Wesley ed. Madrid: 435 p. 0-201-57169-2.

## BIBLIOGRAFÍA

1. POTENCIER, F. y ZANINOTTO, F. *Symfony: la guía definitiva*. 2007, nº Disponible en: <http://www.librosweb.es/symfony>.
2. HEREDIA, M. G. *Los mejores IDEs para Php*. 2007, Disponible en: <http://www.tufuncion.com/ide-php>.
3. RUMBAUGH, J.; JACOBSON, I., *et al. El Proceso Unificado de Desarrollo de Software*. 2000, Disponible en: <http://bibliodoc.uci.cu/pdf/reg00060.pdf>.
4. RUMBAUGH, J.; JACOBSON, I., *et al. El Lenguaje Unificado de Modelado*. 2000, Disponible en: <http://bibliodoc.uci.cu/pdf/reg03050.pdf>.
5. WELLING, L. y THOMSON, L. *Desarrollo Web con PHP y MySQL*. Disponible en: <http://bibliodoc.uci.cu/pdf/reg02819.pdf>.

## GLOSARIO DE TÉRMINOS

**Escalabilidad:** La capacidad del sistema informático de cambiar su tamaño o configuración para adaptarse a las circunstancias cambiantes.

**Redundancia:** Repetición de una información ya dada en el mensaje.

**Rollback:** Es el proceso automático que se ejecuta durante una transacción si un error ocurre la base de datos regresa al estado en que se encontraba desde el última transacción.

**Web Service:** Un servicio Web es una colección de protocolos y estándares que sirve para intercambiar datos entre aplicaciones.

**Scripts:** Un conjunto de comandos escritos en un lenguaje interpretado para automatizar ciertas tareas de aplicación.

**Trazabilidad:** Aptitud de reconstruir la historia, la utilización o la localización de un producto por medio de identificaciones registradas.

**Triggers:** Una acción que causa la invocación automática de un procedimiento, por ejemplo conservar la integridad referencial.

**Tupla:** Es una hilera o fila en una tabla.

**Aplicación Web:** Sitio Web que contiene páginas con contenido sin determinar parcialmente o en su totalidad. El contenido final de estas páginas se determina sólo cuando un visitante solicita una página del servidor Web.

**Base de Datos:** Conjunto de datos almacenados en un soporte informático no volátil. En ella los datos se encuentran interrelacionados y estructurados de acuerdo con su modelo capaz de recoger el máximo de contenido. Se organizan en tablas. Cada fila de una tabla constituye un registro de datos, y cada

columna constituye un campo del registro. Constituyen una parte integrante y fundamental del sistema de información y tiene su razón de ser en la misma existencia de esta.

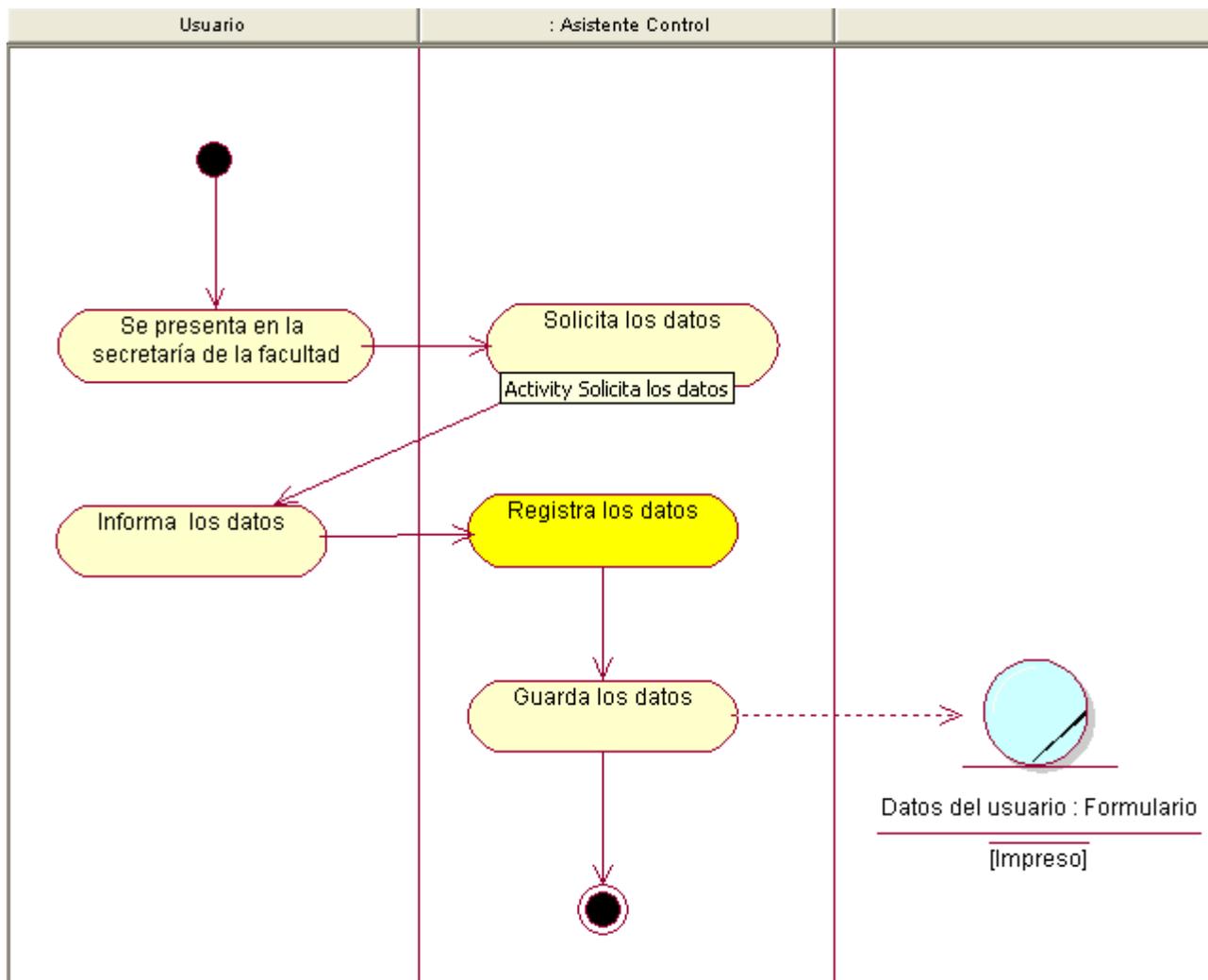
**Servidor Web:** Software que suministra páginas web en respuesta a las peticiones de los navegadores Web.

**Mixin:** En lenguajes de programación orientada a objetos, un mixin es una clase que ofrece cierta funcionalidad para ser heredada por una subclase, pero no está ideada para ser autónoma. Heredar de un mixin no es una forma de especialización sino más bien un medio de obtener funcionalidad. Una subclase puede incluso escoger heredar gran parte o el total de su funcionalidad heredando de uno o más mixins mediante herencia múltiple.

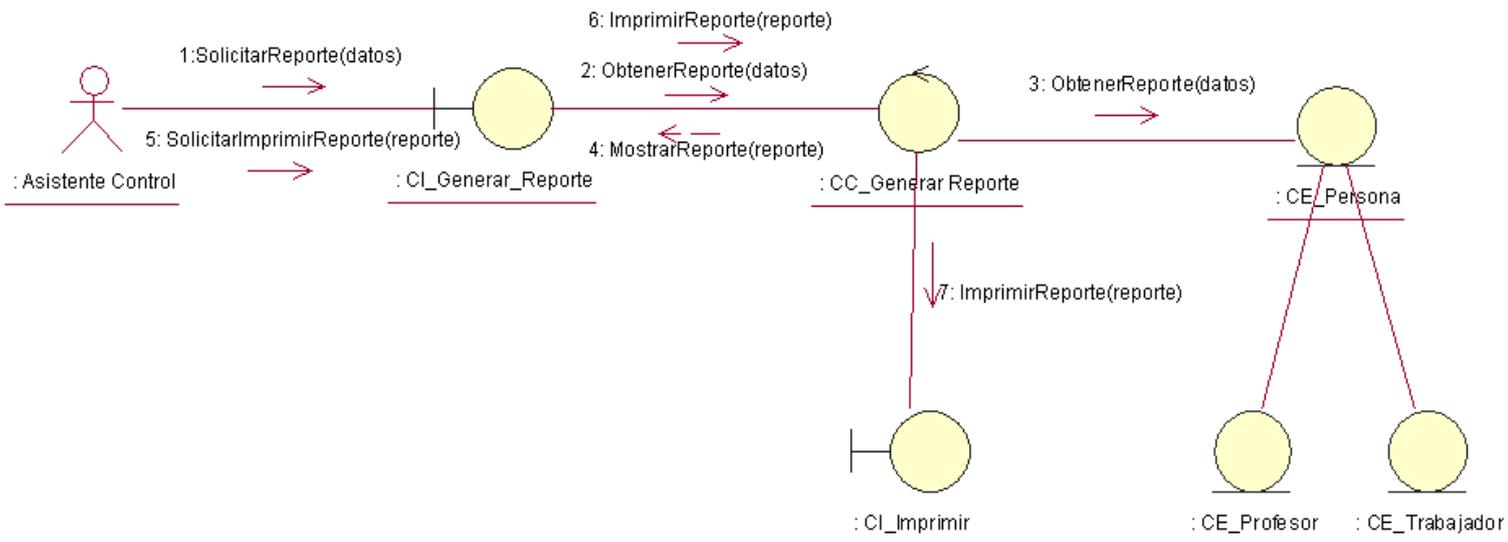
**LDAP (Lighthouse Directory Access Protocol):** Es un protocolo que permite el acceso a un directorio para buscar información que está almacenada en la red. Generalmente la información a que se accede es usuario y contraseña, aunque se pueden tratar otros datos, por lo que se utiliza en general para la autenticación.

**SOAP (Simple Object Access Protocol):** Es un protocolo estándar creado por Microsoft, IBM y otros, está actualmente bajo el auspicio de la W3C que define cómo dos objetos en diferentes procesos pueden comunicarse por medio de intercambio de datos XML. SOAP es uno de los protocolos utilizados en los servicios Web.

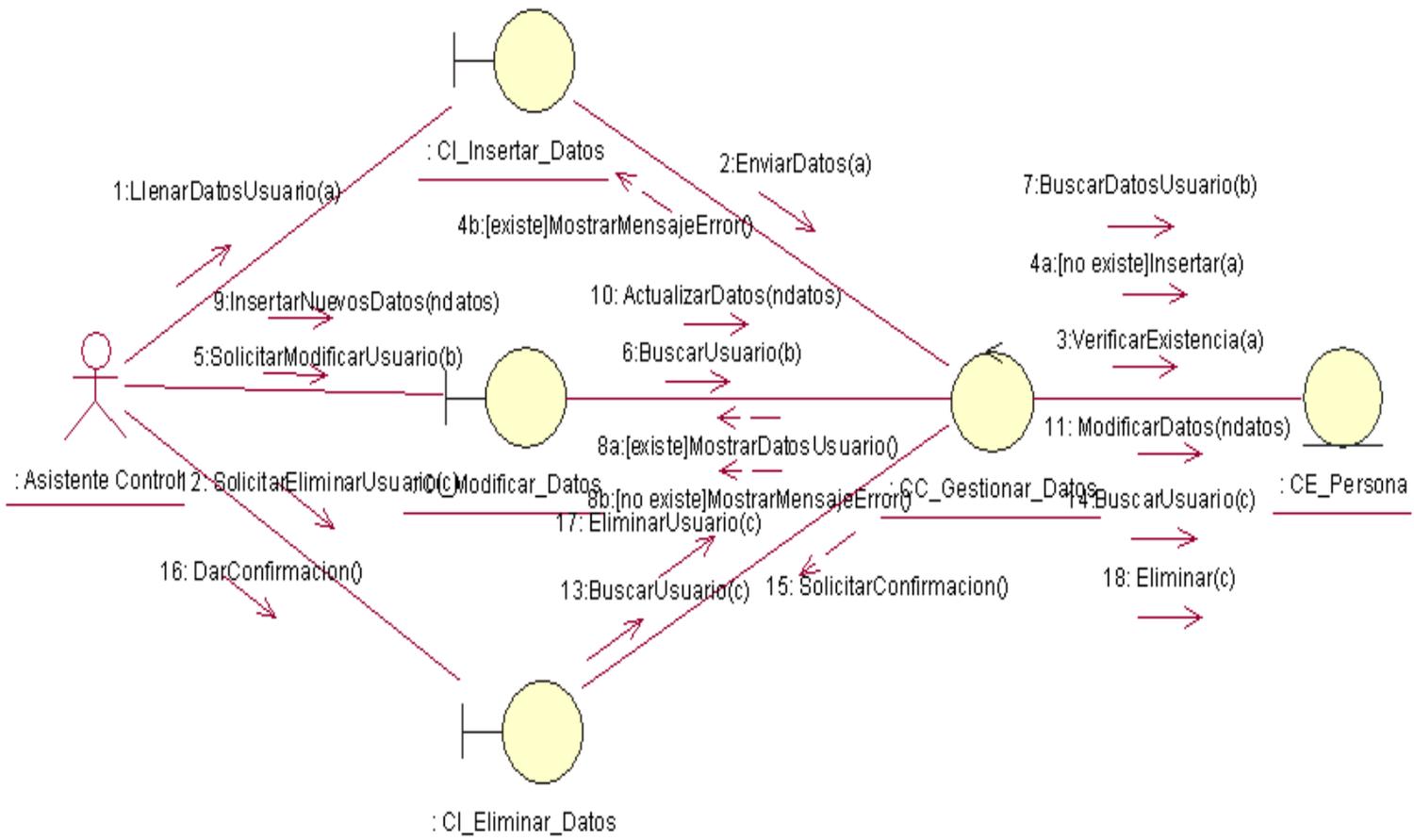
## ANEXOS



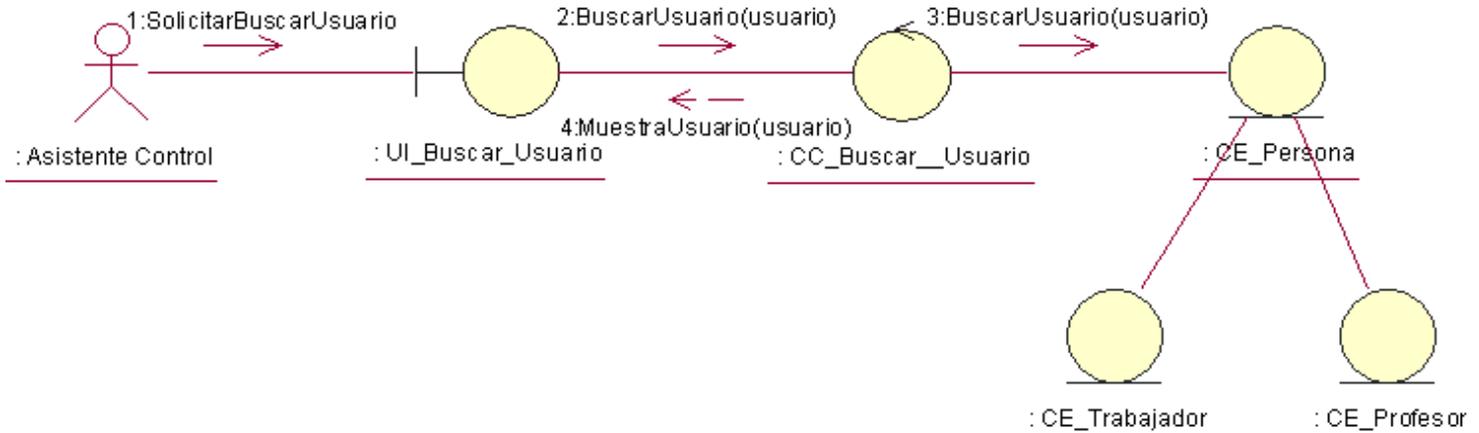
Anexo 1: Diagrama de Actividades del CUN Registrar Datos de Usuario



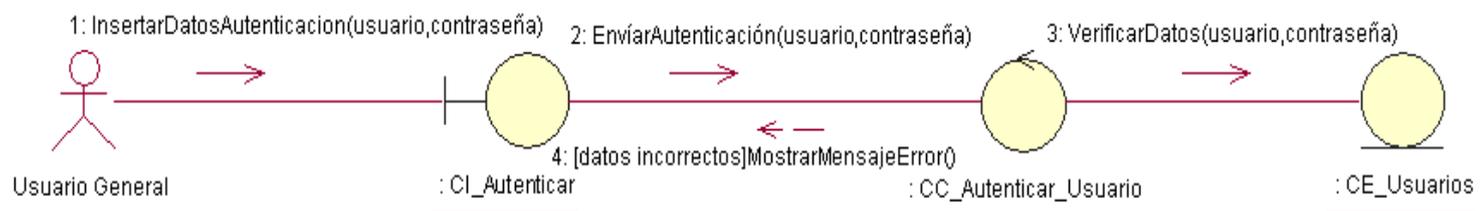
Anexo 2: Diagrama de Colaboración del CUS Generar Reporte



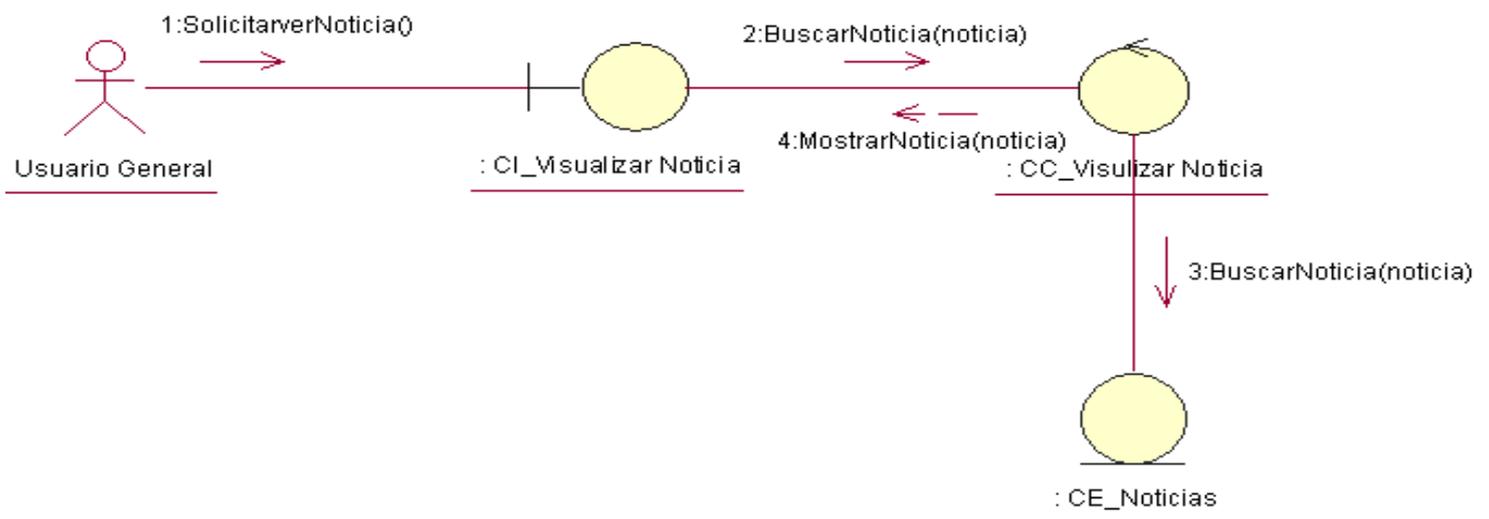
Anexo 3: Diagrama de Colaboración del CUS Gestionar Datos



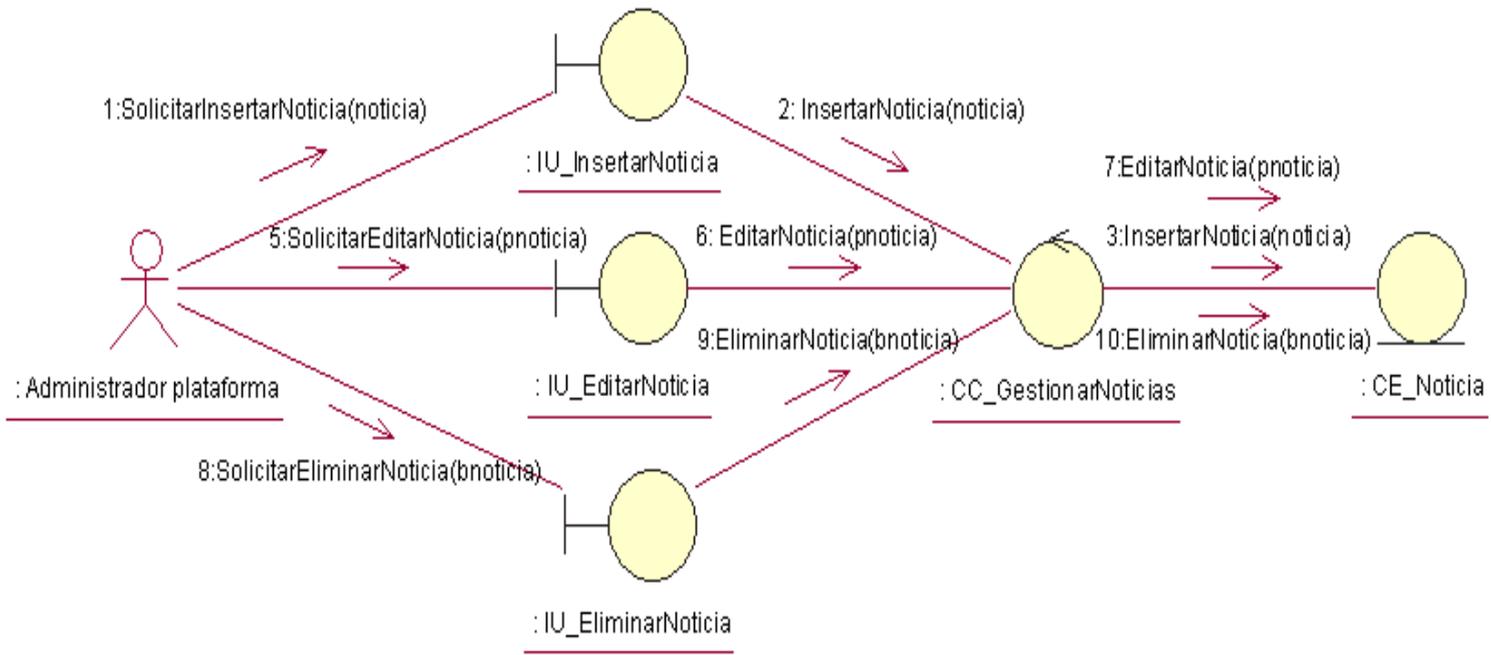
**Anexo 4: Diagrama de Colaboración del CUS Buscar Usuario**



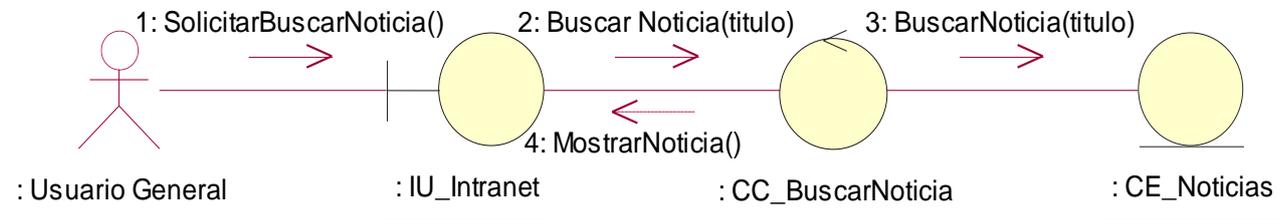
**Anexo 5: Diagrama de Colaboración del CUS Autenticar Usuario**



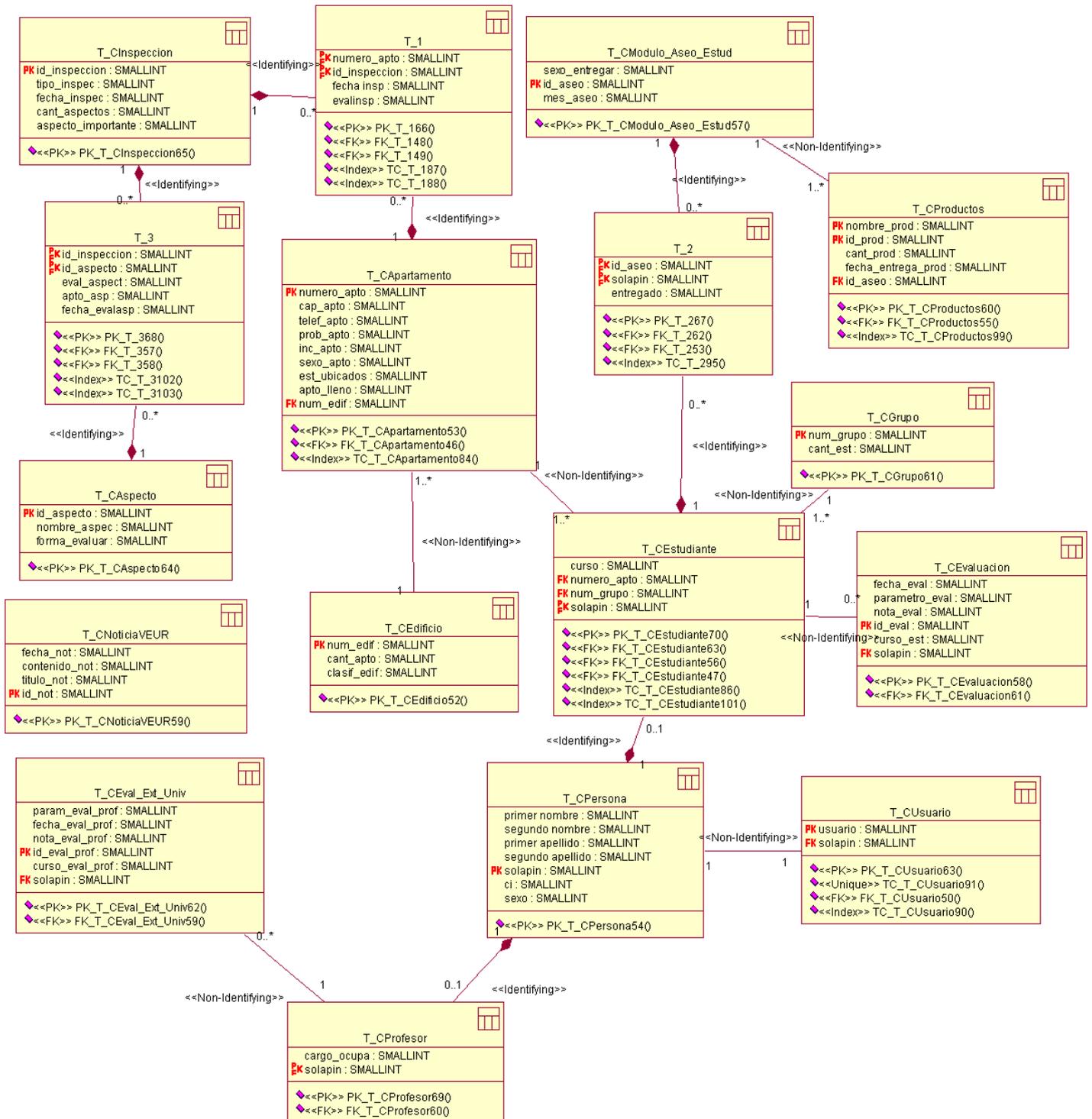
**Anexo 6: Diagrama de Colaboración del CUS Visualizar Noticias**



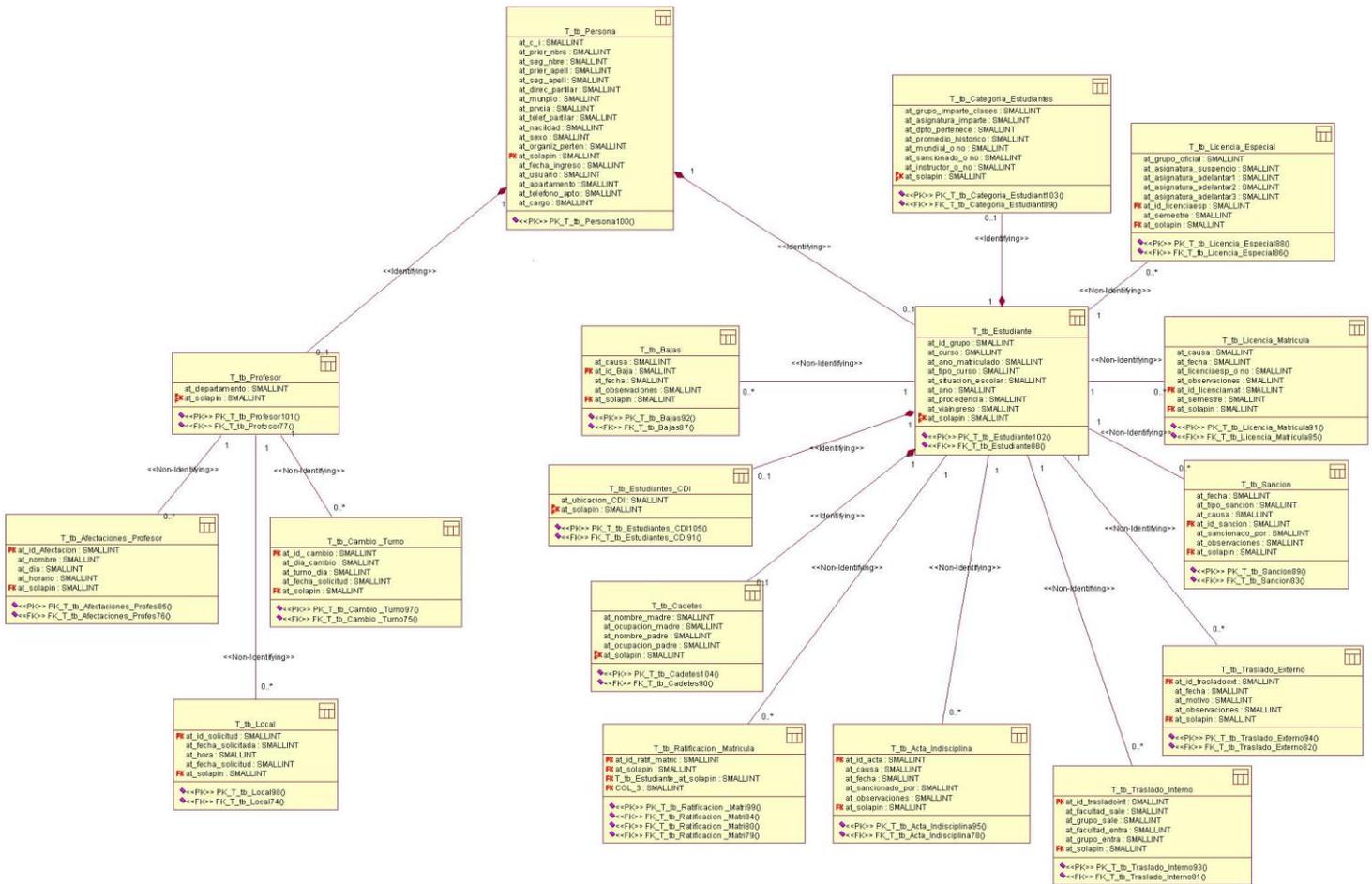
**Anexo 7: Diagrama de Colaboración del CUS Gestionar Noticias**



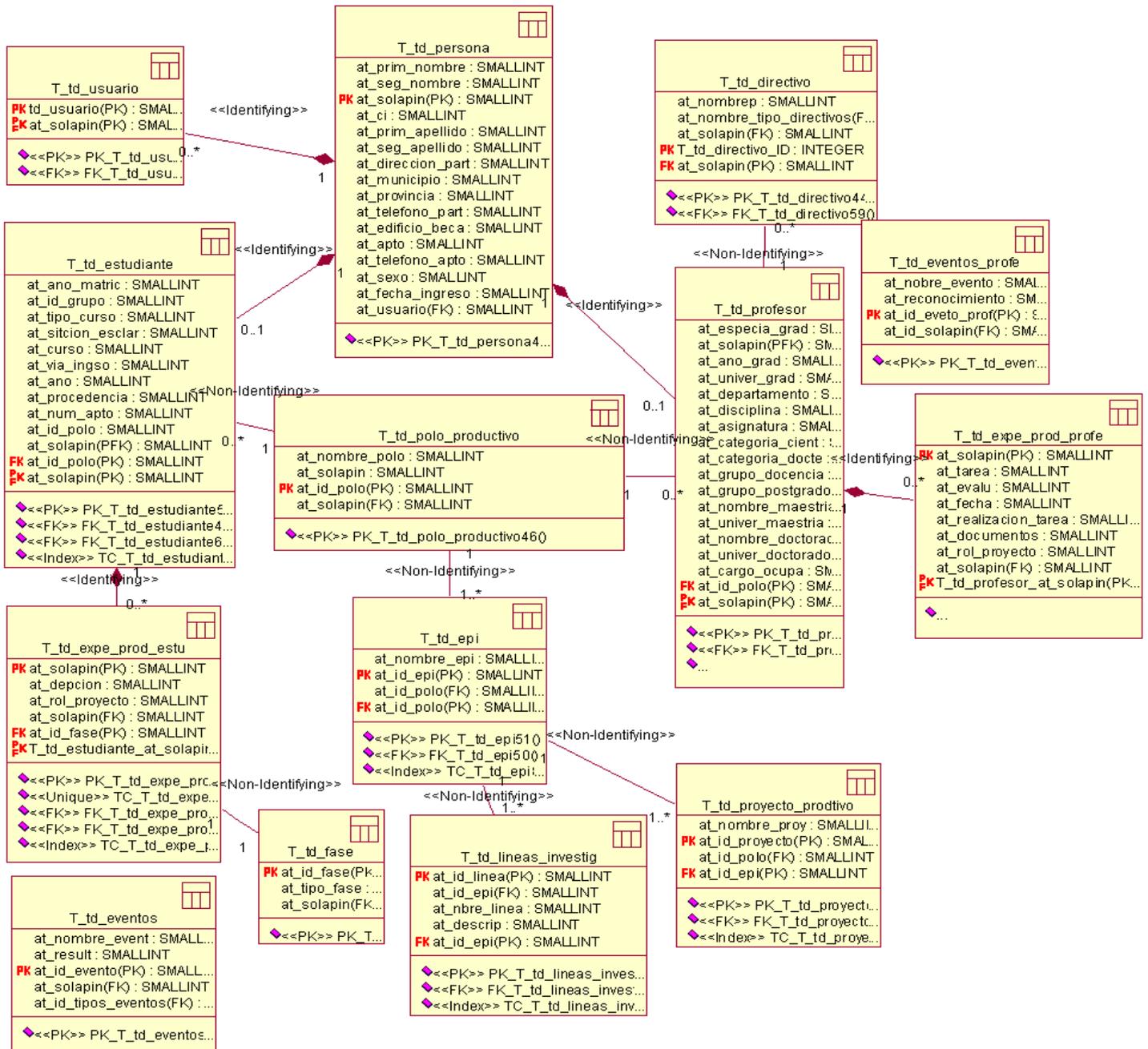
**Anexo 8: Diagrama de Colaboración del CUS Buscar Noticia**



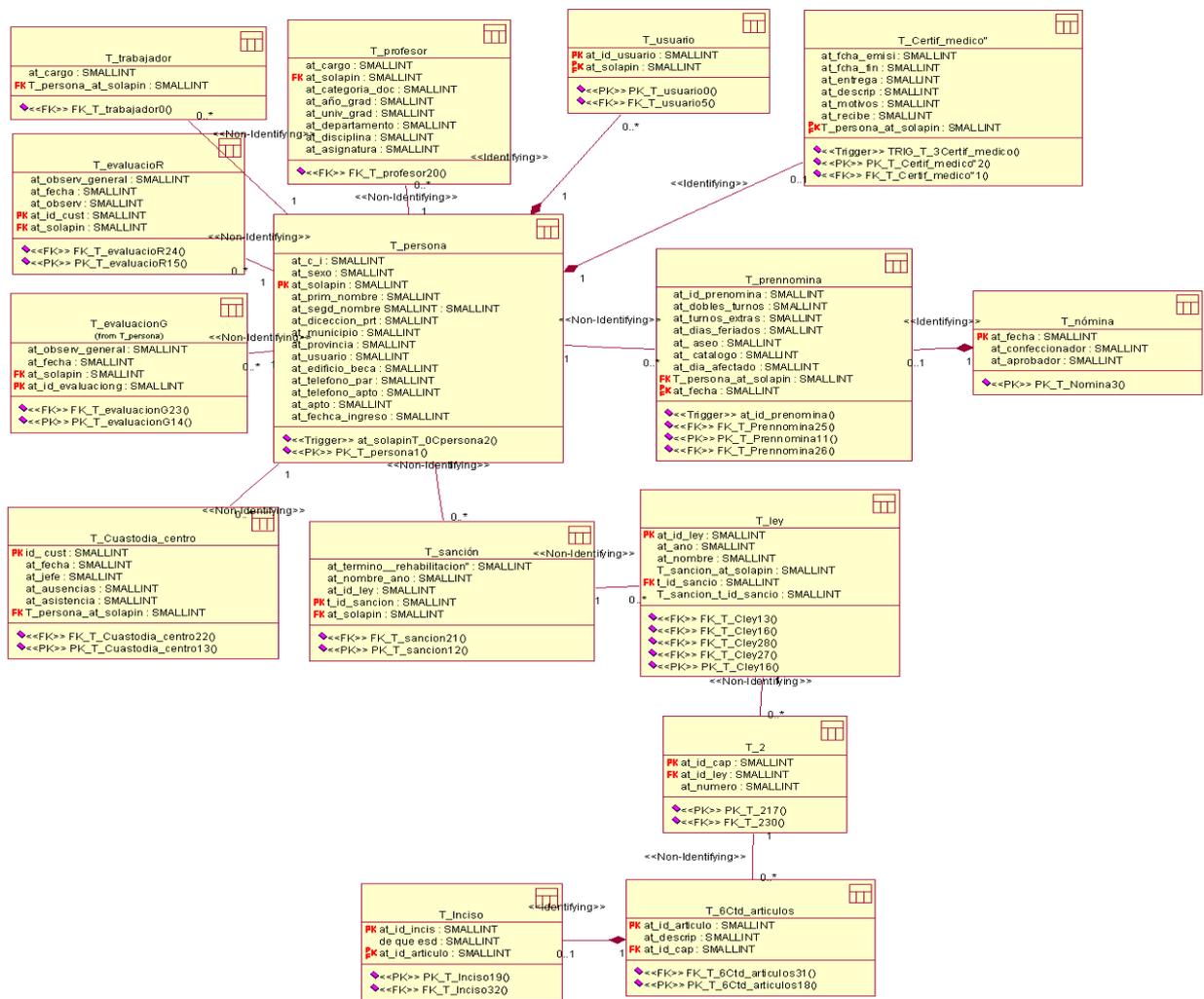
Anexo 9: Modelo de Datos del subsistema de Extensión Universitaria



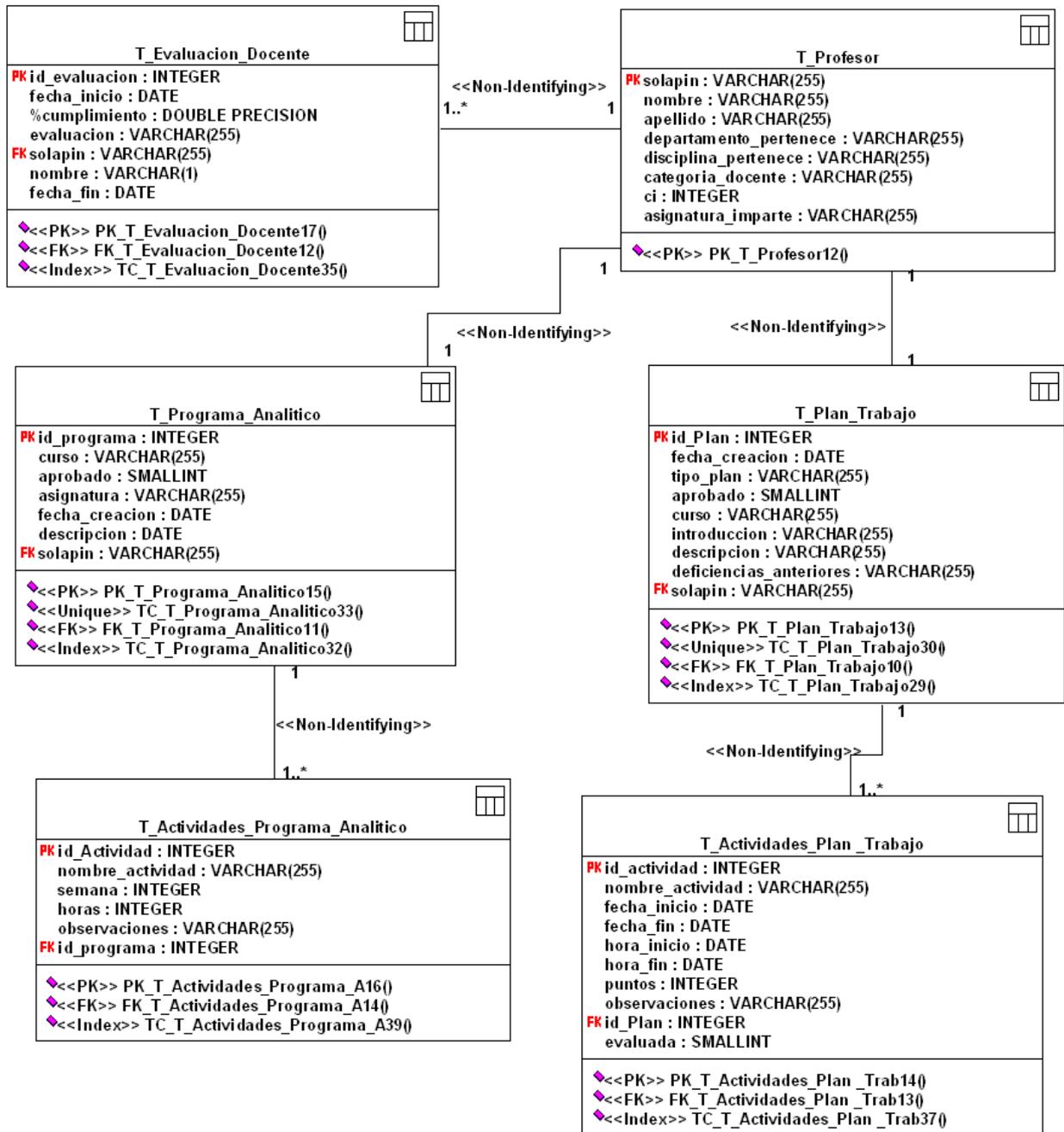
Anexo 10: Modelo de Datos del subsistema de Docencia



Anexo 11: Modelo de Datos del subsistema de Producción



Anexo 12: Modelo de Datos del subsistema de Nómina y Evaluación de Desempeño



Anexo 12: Modelo de Datos del subsistema de Departamento Docente