



**Universidad de las Ciencias Informáticas**

**Facultad 6**



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

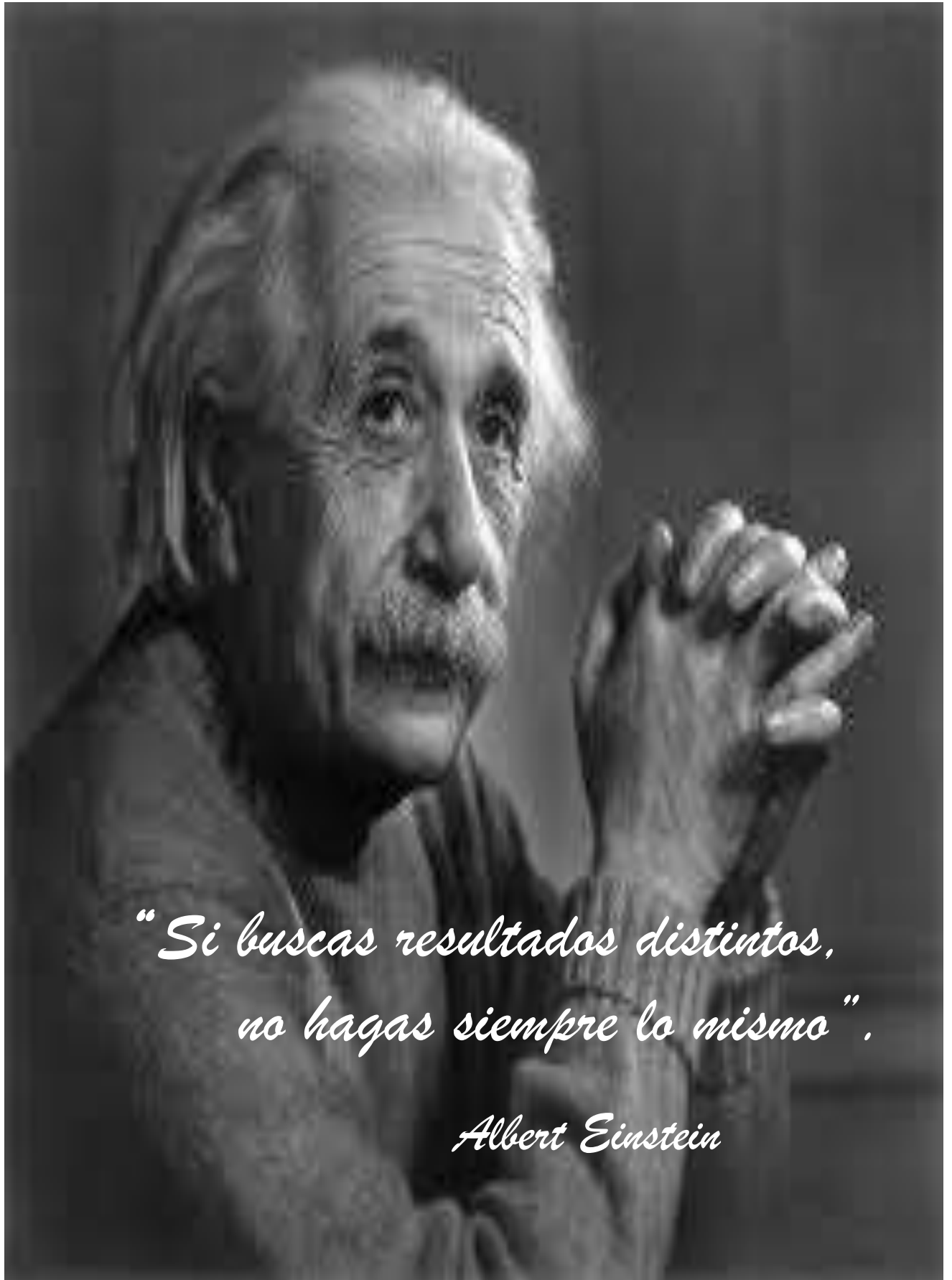
Sistema automatizado para la gestión de la información científico  
técnica del centro GEYSED.

**Autor:** Gabriel David Pérez Fernández

**Tutora:** Ing. Lisbeth O. López Verdecie

La Habana, junio, 2014.

“Año 56 de la Revolución”



*“Si buscas resultados distintos,  
no hagas siempre lo mismo”.*

*Albert Einstein*

### **DECLARACIÓN DE AUTORÍA**

Declaro ser autor del presente trabajo de diploma y autorizo al centro de Geoinformática y Señales Digitales de la Universidad de las Ciencias Informáticas hacer uso del mismo en su beneficio, así como de los derechos patrimoniales con carácter exclusivo.

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

---

Gabriel David Pérez Fernández

Firma del Autor

---

Ing. Lisbeth O. López Verdecie

Firma del Tutor

### DATOS DE CONTACTO

#### Datos del Tutor:

**Nombre y Apellidos:** Ing. Lisbeth Olinda López Verdecie.

**Correo electrónico:** [lolopez@uci.cu](mailto:lolopez@uci.cu)

**Categoría docente:** Asistente

**Año de graduado:** 2008

**Profesión:** Ingeniera en Ciencias Informáticas.

**Breve descripción:** Graduada en la Universidad de las Ciencias Informáticas. Actualmente Jefa del proyecto Sistema de Gestión de Datos Geológicos del Departamento Geoinformática del Centro de Desarrollo “Geoinformática y Señales Digitales” de la facultad 6.

*A mi mamá, la persona más importante de mi vida, espero algún día pueda estar tan orgullosa de mí como yo lo estoy de ella.*

*A mi mamá por apoyarme siempre en todas mis locuras, por confiar en mis decisiones, por complacerme en todos mis caprichos y por todo el amor que me ha brindado... Gracias por ser la mejor mami del mundo.*

*A mi papá y a mi hermano por estar siempre para mí de modo incondicional.*

*A mi tía Luisa por ser mi segunda madre.*

*A mis tías (os) Delia, Marel, Rosita, Hirán, Héctor, Frank y Ricardo por su experiencia y regaños... gracias por construir una familia tan maravillosa.*

*A mis primas(os) Marelita, Claudia, Polo, Rita, Leticia y Luis por su preocupación.*

*A Tatiana, Osmany, Tahími y Tatianita por dejarme formar parte de su familia.*

*A Lisy, mi hermana por su confianza y por los momentos que hemos vivido.*

*A Dainé, mi hermanita malcriada por sus consejos y por soportarme.*

*A Mayi, mi esposa para ti siempre como un boludo con un globo de "I love you" en forma de corazón.*

*A Eddy, por su amistad y por las noches de fiestas.*

*A Yanet, mi amiga y compañera en todo momento.*

*A Sussell, mi hija por su cariño incondicional.*

*A Diatria, Heidis y Arístides por ser excelentes amigos.*

*A todas mis amistades de Holguín, del IPVCE y a los que he conocido en estos últimos años, gracias por esa otra familia que la cual he podido contar en todo momento.*

*A mi tutora por la confianza depositada.*

*A la vida por demostrarme que todo es posible, que todo sucede por un motivo y que en los momentos más difíciles siempre aparece una luz al final del camino.*

*A mí, por todo el esfuerzo de estos últimos meses.*

### **RESUMEN**

En la Universidad de la Ciencias Informáticas (UCI) el desarrollo de acciones asociadas a la investigación y superación de sus profesionales constituye una tarea de primer orden. En los centros de desarrollo de la institución; la información referente a la actividad científica y de postgrado se gestiona a través de indicadores cuantitativos. Específicamente en el centro GEYSED los mecanismos existentes para realizar esta labor no son los más efectivos pues esta información es abundante y se encuentra en diversos formatos y ubicaciones lo cual dificulta el adecuado registro y control de la misma. En la siguiente investigación se describe el desarrollo de una aplicación informática la cual permite gestionar de manera centralizada y homogénea toda la información referente a la actividad científica del centro GEYSED. La implementación del sistema se realiza bajo un ambiente web haciendo uso de tecnologías libres. Las herramientas seleccionadas para el desarrollo del software se integran en un proceso con un enfoque ágil que utiliza las prácticas definidas en la metodología *Xtreme Programming* (XP). Las pruebas realizadas a la aplicación tuvieron resultados satisfactorios, lo que garantiza el correcto funcionamiento del sistema.

### **PALABRAS CLAVE**

Actividad científica, aplicación web, indicadores, postgrado.

## **ABSTRACT**

At the University of Informatics Sciences (UCI) development of actions associated with the investigation and improvement of its professionals is a top priority task. In the development centers of the institution; information regarding scientific and postgraduate activity is managed through scientometric indicators. Specifically in the center GEYSED existing mechanism for this work are not the most effective as this information is abundant and is found in various formats and locations which hinders the proper recording and control of it. In the following research the development of a software application which allows homogeneous and centrally manage all information concerning scientific activity center GEYSED manner described. The system implementation was carried out on a web environment, using open source technologies. These tools are integrated into a software development process to an agile approach using the practices defined in the methodology Xtreme Programing (XP). Testing of the application were successful, ensuring the proper functioning of the system.

## **KEYWORDS**

Indicators, scientific activity, graduate web application.



## Índice de Contenido

INTRODUCCIÓN.....	1
CAPÍTULO 1:Fundamentación teórica de la investigación.....	6
Introducción.....	6
1.1 Conceptos asociados al dominio del problema.....	6
1.2 Gestión de la actividad científico técnica en la UCI.....	7
1.3 Situación problemática .....	9
1.4 Sistemas de gestión de indicadores para medir la Ciencia, Tecnología e Investigación.....	10
1.4.1 Sistema de Gestión de la Investigación (SIG) de la Universidad de Talca, Chile.....	10
1.4.2 Universidad Virtual de Postgrado (IPLAC-Virtual) .....	11
1.4.3 Sistema de Indicadores de Ciencia, Investigación y Tecnología (SIndiCIT).....	11
1.4.4 Sistema de Base de Datos de Indicadores para la Gestión de Información en el área de Investigación y Postgrado de la Facultad 6.....	12
1.5 Conclusiones parciales.....	12
CAPÍTULO 2:Herramientas y tecnologías a utilizar.....	13
Introducción.....	13
2.1 Metodología de desarrollo de software.....	13
2.2 Lenguaje de modelado .....	15
2.3 Herramientas CASE para el modelado.....	16
2.4 Herramientas IDE .....	17
2.5 Lenguaje de programación.....	17
2.5.1 Lenguaje de programación del lado del servidor.....	17
2.5.2 Lenguaje de programación del lado del cliente.....	18
2.6 Framework .....	19
2.7 Sistema Gestor de Bases de Datos.....	20
2.8 Servidor Web.....	21

2.9 Conclusiones parciales.....	22
CAPÍTULO 3:Análisis y diseño de la propuesta de solución. ....	23
Introducción.....	23
3.1 Descripción del sistema.....	23
3.2 Modelo de Dominio.....	24
3.3 Lista de Reserva del Producto(LRP).....	24
3.4 Historias de Usuario .....	26
3.5 Plan de Iteraciones.....	28
3.6 Arquitectura. ....	28
3.7 Patrón de Arquitectura.....	30
3.8 Patrones de Diseño. ....	31
3.9 Tarjetas Clase-Responsabilidad-Colaboración (CRC). ....	33
3.10 Diagrama de Clases .....	34
3.11 Conclusiones parciales.....	35
CAPÍTULO 4:Implementación y pruebas.....	36
Introducción.....	36
4.1 Tareas de la Ingeniería.....	36
4.2 Pruebas.....	38
4.2.1 Prueba de Caja Negra. ....	38
4.3 Conclusiones parciales.....	43
CONCLUSIONES GENERALES.....	44
RECOMENDACIONES .....	45
REFERENCIAS BIBLIOGRÁFICAS.....	46
ANEXOS.....	48

## Índice de Tablas

Tabla 1:Lista de Reserva del Producto .....	25
Tabla 2: HU_2 Gestionar publicaciones de un profesional. ....	27
Tabla 3: Plan de Iteración del sistema .....	28
Tabla 4: Tarjeta CRC Clase Profesional .....	33
Tabla 5: Tarea N°1: Adicionar publicaciones de un profesional.....	36
Tabla 6: Tarea N°2: Modificar publicaciones de un profesional. ....	37
Tabla 7: Tarea N°3: Eliminar publicaciones de un profesional.....	37
Tabla 8: Tarea N°4: Visualizar publicaciones de un profesional. ....	38
Tabla 9: Caso de Prueba: Gestionar publicación (SC_1: Adicionar Publicación) .....	39
Tabla 10: Descripción de variable .....	41
Tabla 11: Matriz de datos (SC_1: Adicionar publicación) .....	42
Tabla 12: HU_1 Gestionar Claustro. ....	48
Tabla 13: HU_3 Gestionar premios de un profesional. ....	49
Tabla 14: HU_4 Gestionar capacitación de un profesional.....	49
Tabla 15: HU_5 Gestionar trabajos de un profesional. ....	50
Tabla 16: HU_6 Gestionar proyectos I+D del centro. ....	50
Tabla 17: HU_7 Gestionar patentes y registros del centro. ....	51
Tabla 18: HU_8 Gestionar resultados introducidos. ....	51
Tabla 19: HU_9 Gestionar categorías. ....	52
Tabla 20: HU_10 Autenticar Usuario.....	52
Tabla 21: HU_11 Gestionar departamento.....	53
Tabla 22: HU_12 Gestionar centro.....	53
Tabla 23: Tarjeta CRC Clase Publicación .....	54
Tabla 24: Tarjeta CRC Clase Trabajo .....	54
Tabla 25: Tarjeta CRC Clase ProyectoID.....	54

## Índice de Figuras

Figura 1: Modelo de Dominio .....	24
Figura 2: Arquitectura Cliente-Servidor.....	30
Figura 3: Patrón Modelo-Vista-Controlador (Potencier, 2008).....	30
Figura 4: Diagrama de clases del diseño asociado a la HU_2 Gestionar publicaciones .....	34
Figura 5: Modelo de datos del sistema. ....	35
Figura.6: No conformidades por iteración.....	43
Figura 7: Diagrama de clases del diseño, HU_1: Gestionar claustro. ....	55
Figura 8: Diagrama de clases del diseño, HU_3: Gestionar premios de un profesional.....	56
Figura 9: Diagrama de clases del diseño, HU_3: Gestionar capacitación de un profesional. ....	56
Figura 10: Diagrama de clases del diseño, HU_5: Gestionar trabajos de un profesional.....	57
Figura 12: Diagrama de clases del diseño, HU_7: Gestionar patentes y registros del centro.....	58
Figura 13: Diagrama de clases del diseño, HU_8: Gestionar resultados introducidos en el centro. ....	58

### INTRODUCCIÓN

La sociedad de la información hace referencia a un nuevo paradigma que ha propiciado la transición de una sociedad que basaba su capital en la producción de bienes de consumo a otra, cuya base de riqueza es la gestión de la información y el conocimiento. La información está presente de manera directa en todos los procesos en las organizaciones actuales, lo cual se traduce en la necesidad de buscar soluciones para el análisis y tratamiento de datos capaces de gestionar de forma fácil, rápida y segura los grandes volúmenes de información que se generan a diario, posibilitando el manejo automatizado y eficiente de estos (Martínez León, 2006).

La digitalización de la información es un proceso que se desarrolla y crece proporcionalmente respecto a los avances alcanzados en las Tecnologías de la Informática y las Comunicaciones (TIC). Este fenómeno está imponiendo a las sociedades modernas la necesidad de elevar el desarrollo de la Ciencia, la Tecnología y la Innovación (CTI), a través del fomento de la investigación, el progreso tecnológico y el fortalecimiento de la competitividad industrial (Sancho, 2010). Cuba no está al margen de estos acontecimientos y se ha planteado la inserción de sus universidades dentro del sistema empresarial, con el objetivo de lograr productos competitivos que satisfagan las necesidades de la economía y la sociedad cubana.

La Universidad de las Ciencias Informáticas (UCI), creada con el objetivo de desarrollar la industria cubana del *software*, constituye un ejemplo fehaciente de lo planteado anteriormente. En esta institución se han desarrollado aplicaciones y servicios informáticos los cuales tributan directamente a la economía del país; además en ella se forman profesionales en la rama de la informática.

La UCI tiene entre sus principales líneas de trabajo la realización de acciones asociadas a la investigación científica, las cuales son planificadas en correspondencia con el capital humano, la infraestructura tecnológica y los recursos materiales y financieros disponibles. La gestión de las actividades de investigación y postgrado en esta institución están guiadas por el Sistema de Indicadores de Ciencia, Tecnología e Innovación (SNCTI), el cual es implementado por el Ministerio de Educación Superior (MES) en las universidades del país. El SNCTI integra un conjunto de indicadores cuantitativos en cinco grupos, calificados cada uno como medidores de la relevancia (premios y reconocimientos obtenidos), la visibilidad (publicaciones científicas), la tecnología (patentes y registros de productos), la pertinencia (recursos financieros ingresados) y el impacto (aporte económico) de los resultados y actividades científico técnicas y de innovación.

La universidad posee una estructura organizativa para el control y seguimiento de las actividades relacionadas con la investigación y la superación; la cual está compuesta entre otros factores por la subdirección de investigación y postgrado, ubicados en cada uno de los centros de desarrollo. El directivo encargado de dirigir esta estructura es el subdirector de investigación y postgrado (SIP), el cual es el responsable de organizar metodológicamente los procesos de capacitación relacionados con la formación científica, gestionar la política científica y aprobar las estrategias de investigación.

En el centro de Geoinformática y Señales Digitales (GEYSED) el subdirector de investigaciones y postgrado elabora un Plan de Indicadores de CTI, en el cual se recoge la proyección del centro para cada año fiscal. A partir de los indicadores planificados se introducen actividades que tributen a la investigación y la superación, dentro del plan de trabajo de los profesionales del centro. El SIP debe recopilar la información que corrobore el cumplimiento de las tareas asignadas; además debe registrar la información asociada a los avances del centro en ciencia y técnica. En determinados períodos del año se realiza el Balance de Ciencia y Técnica (BCT), el cual constituye un resumen del cumplimiento del Plan de Indicadores en un determinado período de tiempo.

Para la ejecución del proceso descrito están creados una serie de mecanismos, pero existen algunas insuficiencias que atentan contra la consistencia y fiabilidad de la información manejada y que ralentizan el proceso de toma de decisiones oportuna. El registro de la información de ciencia y técnica del centro se realiza mediante herramientas ofimáticas, lo cual puede traer implicado la repetición o incongruencias en los datos procesados, pues son diversos los formatos con los cuales se trabaja. El control del desempeño de los profesionales del centro en las tareas asignadas se hace complejo, pues para realizar dicho chequeo se debe acceder al plan de resultados de cada profesional. Las estadísticas que componen el BCT están en diversas ubicaciones y formatos lo cual hace que se torne lento la confección del mismo.

De lo expuesto anteriormente se establece como **problema** a resolver:

¿Cómo contribuir a la organización y control de la información asociada a la actividad científica y de postgrado del centro GEYSED?

El **objeto de estudio** está enmarcado en el proceso de gestión de la información asociada a la actividad científica y de postgrado en la UCI, estableciendo como **campo de acción** la informatización

del proceso de gestión de la información asociada a la actividad científica y de postgrado en el centro GEYSED.

Para la solución del problema planteado se define como **objetivo general** de la investigación: Desarrollar una aplicación informática que permita la organización y control de la información asociada a la actividad científica y de postgrado del centro GEYSED.

Para guiar la investigación se definen las siguientes **preguntas de la investigación**:

- ¿Cuáles son los fundamentos teóricos que sustentan el proceso de gestión de la información asociada a la actividad científica y de postgrado?
- ¿Cuáles son las características que debe cumplir la aplicación informática para que permita la organización y control de la información asociada a la actividad científica y de postgrado del centro GEYSED?
- ¿Cómo estructurar el proceso de desarrollo del sistema informático propuesto de modo que facilite la organización y control de la información asociada a la actividad científica y de postgrado del centro GEYSED?
- ¿Cumple el sistema informático con las funcionalidades previstas?

Para el desarrollo de la investigación se proponen las siguientes **tareas investigativas**:

- Caracterización del proceso de gestión de la información científico y de postgrado en el centro GEYSED, lo cual permite una mejor comprensión del entorno en que se enmarca la investigación y la problemática existente.
- Análisis de las soluciones existentes que responden al problema de la investigación en alguna medida, sus limitaciones y fortalezas.
- Caracterización de las principales herramientas, tecnologías, lenguajes y metodología a utilizar para la construcción de la propuesta de solución.
- Desarrollo del análisis y diseño de la solución propuesta, permitiendo una descripción detallada de cómo implementar el sistema.
- Implementación de la solución propuesta, debido a la necesidad de contar con una aplicación informática que garantice la organización y el control de la información asociada a la actividad científica y de postgrado.

- Desarrollo del proceso de pruebas, con el fin de detectar fallos que puedan ser corregidos para la obtención de un producto que cumpla con las funcionalidades identificadas.

### **Posibles resultados:**

- Una aplicación informática que permita la organización y control de la información asociada a la actividad científica y de postgrado del centro GEYSED.
- La documentación técnica del proceso ingenieril.

### **Métodos Teóricos:**

**Histórico-Lógico:** La utilización de este método facilitó la realización de un análisis de los sistemas de ciencia, tecnología e investigación existentes tanto en el ámbito internacional como nacional, lo cual permitió el entendimiento de sus estructuras y su funcionamiento.

**Análisis-Síntesis:** Se empleó en la revisión de la información consultada sobre los sistemas de gestión de la información científica y postgrado, así como en el estudio del proceso de gestión de las actividades de investigación y postgrado en el centro GEYSED. Lo anterior permitió seleccionar los conceptos, las definiciones y las teorías más importantes relacionadas con el tema.

### **Métodos Empíricos:**

**-Entrevista:** Se efectuaron entrevistas informativas con el subdirector del centro GEYSED el Ing. Jean Michel Suárez. El entrevistador se valió de un cuestionario de preguntas previamente establecido, el cual puede ser consultado en el Anexo I, con el fin de lograr un mejor entendimiento sobre el proceso objeto de estudio. Los encuentros realizados aportaron elementos esenciales para una mejor comprensión del proceso de gestión de la información científico técnica en el centro GEYSED.

El presente trabajo está estructurado por 4 capítulos, distribuido de la siguiente forma:

### **CAPÍTULO 1. Fundamentación teórica de la investigación.**

En este capítulo se ofrece una panorámica acerca de los conceptos y procesos asociados al dominio del problema. Se exponen las principales deficiencias existentes en la gestión de la actividad de investigación y postgrado. Se realiza un análisis del estado del arte de los sistemas existentes a nivel internacional y nacional aplicados a las área de investigación y postgrado.



## **CAPÍTULO 2. Herramientas y tecnologías a utilizar.**

En el presente capítulo se presenta la metodología que guía el ciclo de vida del *software*. Se definen el entorno tecnológico y los lenguajes de programación a utilizar para el desarrollo de la solución propuesta.

## **CAPÍTULO 3. Análisis y diseño de la propuesta de solución.**

En este capítulo se muestra una propuesta de solución al problema planteado, así como los requisitos que debe cumplir el *software*. Se detallan los artefactos asociados a la metodología seleccionada para guiar el proceso de desarrollo de la aplicación. Se especifica la arquitectura, patrones arquitectónicos y de diseño utilizados y se representan las clases del sistema y sus relaciones.

## **CAPÍTULO 4. Implementación y pruebas.**

En el presente capítulo se visualiza la interacción entre los componentes del sistema. Se muestran las Tareas de Ingeniería para lograr una mejor comprensión de la fase de implementación. Se valida la correcta implementación de las HU definidas mediante la aplicación de casos de pruebas, como premisa del adecuado funcionamiento del *software*.

# CAPÍTULO 1

## Fundamentación teórica de la investigación.

### Introducción

En este capítulo se abordan los elementos relacionados con los fundamentos teóricos, los cuales sirven como sustento para la presente investigación. Se realiza una descripción de los principales conceptos y procesos asociados a la gestión de la actividad de investigación y superación en el centro GEYSED. Se detalla el resultado del estudio del estado del arte realizado sobre los sistemas relacionados con la gestión de postgrados y la actividad científica, tanto a nivel nacional como internacional.

### 1.1 Conceptos asociados al dominio del problema.

En la UCI las acciones asociadas a la actividad investigativa y de superación de sus profesionales se ven reflejadas en otros aspectos a través de las publicaciones, trabajos realizados y premios obtenidos. Estas actividades son medidas a través de indicadores cuantitativos, los cuales permiten evaluar el progreso alcanzado en cuanto a CTI por la universidad.

La **cuantimetría** es la disciplina que estudia la producción científica con el fin de medir y analizar la ciencia. Se encarga de analizar factores que pueden ser determinantes para el desarrollo de la actividad científica y de la evaluación de la producción científica mediante indicadores numéricos de publicaciones, patentes, además examina el desarrollo y las políticas científicas (Spinak, 1998).

Los **indicadores** son herramientas para clarificar y definir, de forma más precisa, objetivos e impactos. Son medidas verificables de cambio o resultado diseñada para contar con un estándar contra el cual evaluar, se utilizan principalmente para estimar o demostrar el progreso con respecto a metas establecidas (Pérez Mondragón, 2002). Los indicadores para la gestión de la información de CTI no son más que parámetros que se utilizan para la evaluación de las actividades de investigación e innovación, siendo utilizados principalmente para orientar cómo se puede obtener mejores resultados (Arjonilla, 2009).

Tomando como referencia el criterio de los autores citados y en el marco del presente trabajo se define como un indicador un valor numérico que permite medir o evaluar cuantitativamente el nivel alcanzado

en relación a la actividad científica en una determinada área a partir del cumplimiento de actividades asociadas a la ciencia, investigación y la superación.

La realización de cursos de **postgrado** también forma parte de la actividad científico técnica de la universidad; estos constituyen *“los estudios de especialización luego de haber cursado la licenciatura o pregrado, forman parte del tercer ciclo de estudio de la formación educacional. Se entiende como estudios de postgrado los diplomados, maestrías y doctorados los cuales se realizan con el objetivo de elevar la categoría científica del egresado de pregrado.”* (Acreditación, 2013)

### **1.2 Gestión de la actividad científico técnica en la UCI.**

La adecuada gestión de la ciencia y la tecnología constituye un factor decisivo en el desarrollo de los diferentes sectores de la economía y la sociedad. La construcción de indicadores capaces de medir el impacto, la eficacia y eficiencia de las actividades relacionadas con la ciencia y tecnología, recibe cada vez más atención por parte de las instituciones del país.

La UCI, como centro de educación superior adscrito al MES mide en cada una de sus áreas, el desempeño de la actividad científica y de postgrado a través del SNCTI. Respecto a la aplicación de este sistema en la UCI el autor Yordanis Medina León plantea en su tesis de maestría Sistema informático para la gestión de los indicadores de Ciencia e Innovación Tecnológica *“dicho sistema no se adapta en su totalidad a las condiciones existentes en la universidad, pues la misma se diferencia del tipo de universidad clásica del país; la formación curricular y núcleo fundamental de las investigaciones que se desarrollan se centran en las ramas de la Ciencias de la Informática y la Computación. La universidad se propone participar de manera activa en la producción de software enfocados en la informatización de la sociedad y la exportación de productos informáticos con alto valor agregado para la economía del país”* (Medina León, 2012).

Por lo planteado anteriormente no se hace factible la aplicación en su totalidad del SNCTI, entre las principales dificultades se destacan (Medina León, 2012) :

- Los índices definidos no diferencian las investigaciones que se realizan en las Ciencias Informáticas y de la Computación de las vinculadas a otras ramas del conocimiento.
- La definición del indicador Resultados Introducidos no es clara pues no diferencia entre un producto de aplicación restringida, con relación a sistemas robustos de amplio espectro e impacto económico y social.

- No tiene en cuenta la calidad del proceso de formación del Capital Humano.

Para la utilización de los indicadores propuestos en el SNCTI, los mismos han sido modificados según las estrategias e intereses de la universidad en el área de investigaciones y postgrado. Como resultado de lo anterior en la UCI son empleados los siguientes indicadores (Indicadores, 2009):

- **Premios obtenidos** (internacionales, nacionales, otorgados por la Academia de Ciencias de Cuba, en el Fórum Nacional de Ciencia y Técnica, por el CITMA, en el Concurso Nacional de BTJ y Exposición Forjadores del Futuro, por adiestrados o reservas en el Fórum Nacional de Ciencia y Técnica y Sellos Forjadores del Futuro).
- **Publicaciones científicas** ( revistas referenciadas en la *Web of Science* del Instituto de Información Científica (ISI), revistas científicas referenciadas en base de datos reconocida internacionalmente, revistas científicas nacionales arbitradas, libros publicados en Cuba y el extranjero, en Memorias de Eventos, serie internas y repositorios institucionales).
- **Patentes y registros** (patentes de invención y modelos de utilidad solicitados en Cuba y en el extranjero, patentes de invención y modelos de utilidad concedidos en Cuba y en el extranjero, registro de productos, equipos, medios no informáticos y *software* en Cuba y el extranjero).
- **Participación en proyectos I+D** (nacionales, territoriales, ramales, empresariales y universitarios).
- **Resultados introducidos** (por productos culminados en proyectos o módulos independientes en proyectos grandes, medianos y pequeños).
- **Trabajos presentados en eventos** (internacionales en el extranjero o realizados en Cuba, aceptados y no expuestos en eventos internacionales, aceptados y no expuestos en eventos nacionales e ingresos por eventos realizados en el centro).
- **Capacitación**
  - Recibida** (diplomados, maestrías y doctorados cursando o concluidos).
  - Ofertada** (claustros de maestrías, diplomados, doctorados curriculares, postgrados y conferencias científicas impartidas).

- **Uso de estudiantes** (ponencias en eventos internacionales, nacionales y a nivel UCI; estudiantes premiados en concursos internacionales, nacionales y a nivel UCI, copas y concursos y vinculados a trabajos de investigación o I+D).

El proceso de gestión de la actividad científica técnica en el centro GEYSED parte de un Plan de Indicadores para cada año fiscal. Conocidos los indicadores a medir se planifican tareas a cumplir por los profesionales del centro, corroborándose su ejecución una vez realizadas. En determinados períodos del año se realizan resúmenes estadísticos en base al cumplimiento de los indicadores propuestos constituyendo estos los elementos que componen el Balance de Ciencia y Técnica.

### 1.3 Situación problemática

El correcto desarrollo del proceso de gestión de la actividad de investigación y superación en el centro GEYSED se ve afectado por una serie de deficiencias en los mecanismos creados para su ejecución, los cuales se reflejan a continuación:

- El subdirector de investigaciones y postgrado gestiona todo su contenido de trabajo de forma manual tornándose lento y complejo, entre las causas que inciden en esta situación están: la cantidad y diversidad de formatos (.doc, .xls y .pdf) en la que se encuentra la información, la poca disponibilidad de la misma y en ocasiones los datos pueden estar desactualizados o ser inconsistentes. Estas dificultades inciden de forma negativa en el momento de informar a las diferentes estructuras de dirección de la universidad sobre el cumplimiento de los indicadores previstos en una etapa determinada.
- Resulta complejo el seguimiento y registro de la actividad investigativa de los profesionales del centro. La revisión de esta información se realiza accediendo a cada uno de los planes de resultados individuales de los implicados lo cual atenta contra la agilidad con que se debe desarrollar dicho proceso.
- El modo en que se realiza el control de las actividades de investigación y postgrado no permite obtener estadísticas fiables sobre el progreso de estas en determinados períodos de tiempos.

### 1.4 Sistemas de gestión de indicadores para medir la Ciencia, Tecnología e Investigación.

El desarrollo alcanzado en cuanto a ciencia, tecnología e investigaciones de instituciones o naciones ha llevado a la necesidad de crear mecanismos capaces de medir el desarrollo e impacto de estas actividades. En la actualidad existen sistemas informáticos que permiten la gestión de la actividad científico técnica, los cuales facilitan el monitoreo y control de la misma. En el presente acápite se realiza un estudio del arte de sistemas que en cierta medida pueden constituir soluciones al problema planteado en la presente investigación.

#### 1.4.1 Sistema de Gestión de la Investigación (SIG) de la Universidad de Talca, Chile.

El SIG fue desarrollado en aras de apoyar la investigación que realizan los académicos de la Universidad de Talca, además constituye una vía de encuentro entre la oferta investigativa de la universidad y la demanda de investigación de la sociedad y las empresas (Scielo, 2010). El sistema se estructuró en base de una gama de servicios de información los cuales son de utilidad exclusiva para determinados usuarios en base a su perfil, entre los que se encuentran:

- Actualización de programas y proyectos de investigación, proyectos de tesis y sus consiguientes resultados, sean estos, publicaciones, presentaciones en congresos disciplinarios, organización de eventos, u otros.
- Información sobre los estados en que se encuentran los proyectos y la disponibilidad de fondos concursales internos.
- Despliegue de indicadores de gestión asociados a las capacidades y resultados de la investigación que se desarrolla en la Universidad de Talca.

Este sistema fue desarrollado bajo la plataforma tecnológica Oracle *Workflow* versión 2.6 y PHP 4.3.6, además cuenta con una serie de subsistemas integrados a él, como son Sistema de Gestión de la Universidad (SGU), el Subsistema de Gestión Administrativo y Financiero (SAFI) y el Subsistema de Gestión Académica (SAGA) para establecer la cooperación del SGI con los demás subsistemas se hace uso *Web Services*.

El sistema de Gestión de la Investigación de la Universidad de Talca, Chile posee una buena organización en cuanto a servicios relacionados con el área de la investigación, la superación y demás

áreas de la universidad a partir de los subsistemas que se le fueron integrados. El SIG no es una solución factible, pues fue desarrollado para satisfacer un conjunto de necesidades o deficiencias de mayor nivel que las detectadas al hacer el análisis de la situación problemática en esta investigación, pero el estudio de su estructura y de su componente informativo si posee gran importancia para el desarrollo de la propuesta de solución a construir.

### ***1.4.2 Universidad Virtual de Postgrado (IPLAC-Virtual)***

El Instituto Pedagógico Latinoamericano y Caribeño (IPLAC) con sede en La Habana está adscrito a la Universidad de Ciencias Pedagógicas Enrique José Varona. El IPLAC contribuye al desarrollo de las investigaciones pedagógicas, sociológicas y los estudios de la educación comparada en América Latina y el Caribe. Ofrece diversos cursos, programas específicos para profesores y directivos de la educación en temas de Ciencias Pedagógicas y Ciencias de la Educación. Los encuentros se desarrollan de manera presencial o a distancia con el apoyo de las Tecnologías de la Información y la Comunicación.

Esta institución ofrece la posibilidad de realizar doctorados, maestrías en Ciencias de la Educación, diplomados, pasantías, investigaciones pedagógicas, asesorías y consultorías. El IPLAC fomenta el intercambio profesional y encuentros bilaterales y la realización de proyectos educativos e investigativos. En el portal está publicada la información referente a los diferentes postgrados en ofertas organizados por unidades temáticas y facilita la consulta de los datos profesionales de los docentes integrados a esta Universidad Virtual.

El IPLAC no se adecua a las necesidades específicas planteadas en la presente investigación pues está enfocada a las actividades de postgrado asociadas con temas relacionados con las Ciencias de la Educación. El portal no posee un registro de los estudiantes matriculados en los cursos ofertados y no permite subir o descargar documentos asociados a los postgrados.

### ***1.4.3 Sistema de Indicadores de Ciencia, Investigación y Tecnología (SIndiCIT).***

El Sistema de Indicadores de Ciencia, Investigación y Tecnología (SIndiCIT) existente en la UCI, brinda la posibilidad de llevar un control de los indicadores relacionados con la ciencia, tecnología e innovación que se especifican hoy en día en cada una de los centros que componen las facultades. Es un sistema de indicadores que permite evaluar la producción científica de los profesores, investigadores y estudiantes de la Universidad, que potencia los resultados científicos y de innovación, se adapta a las características de la universidad. Este sistema contribuye a ponderar de manera

diferenciada aquellas investigaciones científicas de ciclo completo (Investigación + Desarrollo + Producción + Comercialización) con relación a aquellas investigaciones puramente académicas.

El SIndiCIT facilita el control mediante indicadores del desempeño en la actividad científico técnica, lo cual es un elemento que deber ser tomado en cuenta como parte de la solución del problema planteado en la investigación. Posee como agravante que permite consultar el cumplimiento de los indicadores por áreas imposibilitando el análisis del desempeño de los profesionales de modo individual.

### ***1.4.4 Sistema de Base de Datos de Indicadores para la Gestión de Información en el área de Investigación y Postgrado de la Facultad 6.***

Este sistema de base de datos fue desarrollado en el año 2012, fue diseñado para gestionar de forma centralizada la información relacionada con las investigaciones y postgrados de los profesores y especialistas de la facultad 6. El sistema se apoya en el Generador de Reportes Dinámicos 1.7, desarrollado en el centro DATEC de la Universidad de las Ciencias Informáticas, lo cual posibilita la elaboración de reportes asociados con las categorías docentes y científicas de los profesionales, participación en eventos, publicaciones realizadas, premios obtenidos así como generación de reportes en períodos especificados.

El Sistema de Base de Datos de Indicadores para la Gestión de Información en el área de Investigación y Postgrado de la Facultad 6 permite el registro y control de un conjunto de indicadores relacionados con las publicaciones, capacitación y premios obtenidos por un profesional, pero no permite la gestión de otros indicadores como las patentes y registros, proyectos I+D y resultados introducidos los cuales están vinculados directamente con la producción y son de suma importancia para gestión de la actividad científico técnica del centro GEYSED.

## **1.5 Conclusiones parciales**

En el presente capítulo se estableció la base teórica necesaria para facilitar la comprensión de la propuesta del sistema para la gestión de la actividad de investigación y postgrado del centro GEYSED. El análisis de los conceptos y la descripción de los procesos asociados a la gestión de la información de investigación y postgrado aportaron elementos claves para una mejor comprensión de la presente problemática, las partes involucradas y las conexiones existentes entre estas. Los sistemas analizados no constituyen soluciones factibles a la situación planteada, pero el concomitamiento de los mismos constituye una importante base para el desarrollo de la aplicación a construir.



# CAPÍTULO 2

---

## Herramientas y tecnologías a utilizar.

### Introducción

El desarrollo de una solución informática puede ser una tarea compleja, pues son diversos los lenguajes, las metodologías y herramientas que pueden ser utilizados en este proceso. La obtención de un producto de *software* eficiente y que satisfaga las expectativas de los usuarios finales dependerá en parte de la selección que se haga de las herramientas y tecnologías las cuales deben estar acorde con las características y necesidad que se desean satisfacer. En el presente capítulo se realiza el análisis y la selección de las herramientas, lenguajes y metodología que se emplean en el desarrollo del sistema para informatizar la gestión de la información científico técnica del centro GEYSED.

### 2.1 Metodología de desarrollo de software.

Las metodologías son el conjunto de procedimientos, técnicas, herramientas y un soporte documental, que ayuda a los desarrolladores a realizar un nuevo *software*. Existen diversas metodologías de desarrollo de *software*, estas se pueden clasificar en: metodologías ágiles y tradicionales o robustas. Las metodologías ágiles son utilizadas cuando el desarrollo de *software* es incremental (entregas pequeñas de *software*, con ciclos rápidos), cooperativo (cliente y desarrolladores trabajan juntos constantemente con una cercana comunicación) y adaptable (permite realizar cambios de último momento).

#### XP-Xtreme Programing

La Programación Extrema, XP por sus siglas en inglés, es una metodología ligera de desarrollo de *software* que se basa en la comunicación continua entre el cliente y el equipo de desarrollo, comunicación fluida entre todos los participantes, simplicidad en las soluciones implementadas y flexibilidad para enfrentar los cambios (Calabria, 2003). La Programación Extrema se basa en los siguientes conceptos:

- Los equipos de desarrollo trabajan directamente con el cliente durante ciclos cortos de una o dos semanas como máximo.

- La entrega de las versiones del *software* ocurre muy temprano y en intervalos muy cortos para maximizar la interacción con el usuario.
- Existe una fuerte colaboración entre el equipo de desarrollo mientras trabaja en el código.
- El código se prueba y depura a lo largo del proceso de desarrollo.

Esta metodología se define en las siguientes fases:

### **Planificación**

En esta fase se recopilan los requisitos que permiten que los miembros equipo de desarrollo entiendan el contexto del negocio y conozcan las características y funcionalidades del *software*. Los clientes definen las Historias de Usuarios y asignan un valor (es decir, una prioridad) a las historias con base a el valor general de la característica o función para el negocio. Los clientes y desarrolladores trabajan juntos para decidir cómo agrupar las historias ordenándolas en el modo en que serán desarrolladas, definiéndose de este modo los planes de Iteración o entrega. XP asume que la planificación variará en función de cómo varíen las necesidades del negocio.

### **Diseño**

XP hace uso de tarjetas Clase-Responsabilidad-Colaborador (CRC) como un mecanismo eficaz para analizar el *software* en un contexto orientado a objetos. Las tarjetas CRC identifican y organizan las clases orientadas a objetos que son relevantes para el incremento actual de *software*. XP estimula el rediseño, técnica de construcción que también es un método para la optimización del diseño. En XP para realizar el diseño no se utiliza notación y se generan pocos artefactos, por lo cual es una etapa de transición que puede y se debe modificar continuamente a medida que avanza la construcción.

### **Codificación**

Después de que las historias han sido desarrolladas y de que se ha hecho el trabajo de diseño preliminar, el equipo no inicia la codificación, sino que desarrolla una serie de pruebas unitarias a cada una de las historias que se van a incluir en la entrega en curso (incremento de *software*). Una vez creada la prueba unitaria, el desarrollador está capacitado para centrarse en lo que debe implementarse para pasar la prueba.

### **Pruebas**

La creación de pruebas unitarias antes de que comience la codificación es un elemento clave del enfoque de XP. Las pruebas unitarias que se crean deben implementarse con el uso de una estructura

que permita automatizarlas (de modo que puedan ejecutarse en repetidas veces y con facilidad). Las pruebas de aceptación XP, también llamadas pruebas del cliente son especificadas por el cliente y se centran en las características y funcionalidades generales del sistema que son visibles y revisables por parte del cliente. Las pruebas de aceptación se derivan de las Historias de los Usuarios que se han implementado como parte de la liberación del *software*.

Se selecciona la metodología XP para guiar el proceso de desarrollo del sistema, pues la misma está pensada para equipos de desarrolladores pequeños y un ciclo de desarrollo corto. Genera pocos artefactos y no establece un contrato tradicional o al menos es bastante flexible con el cliente, al cual se le puede hacer pequeñas entregas del sistema pues se cuenta con un corto plazo de tiempo para la entrega del producto terminado.

### 2.2 Lenguaje de modelado

El lenguaje de modelado es un conjunto estandarizado de símbolos para modelar parte de un diseño de *software* orientado a objetos. Se utiliza en combinación con una metodología de desarrollo para avanzar de una especificación inicial a un plan de implementación y para comunicar dicho plan a todo el equipo de desarrolladores.

#### UML 2.0

UML" son las siglas de Lenguaje Unificado de Construcción de Modelos (*Unified Modeling Language*), define la manera estándar en que intercambian criterios los miembros del equipo de desarrollo de *software*. Es un estándar, que tiene como objetivo proveer un lenguaje consistente para:

- Visualizar: expresar de una forma gráfica un sistema de forma que otro lo puede entender.
- Especificar: especificar cuáles son las características de un sistema antes de su construcción.
- Construir: A partir de los modelos especificados se pueden construir los sistemas diseñados.
- Documentar: Los propios elementos gráficos sirven como documentación del sistema desarrollado que pueden servir para su futura revisión.

Se decidió emplear UML porque facilita la modelación de sistemas con tecnología orientada a objetos y recomienda la utilización de diagramas para representar las distintas vistas de un sistema. Sus diseños se pueden implementar en cualquier lenguaje de programación que soporte las posibilidades de UML,

permite generar código a partir de los modelos y a la inversa (a partir del código fuente generar los modelos) y tener actualizado la estructura del proyecto con una visión en el diseño de más alto nivel.

### 2.3 Herramientas CASE para el modelado.

Las Herramientas de Ayuda al Desarrollo de Sistemas de Información, surgieron para intentar dar solución a los problemas inherentes a los proyectos de generación de aplicaciones informáticas: plazos y presupuestos incumplidos, insatisfacción del usuario, escasa productividad y baja calidad de los desarrollos. Algunas de estas herramientas se dirigen principalmente a mejorar la calidad, como es el caso de las herramientas CASE (acrónimo de *Computer Aided Software Engineering*). Las herramientas CASE son un conjunto de métodos, utilidades y técnicas que facilitan la automatización del ciclo de vida del desarrollo de sistemas de información, completamente o en alguna de sus fases.

#### Visual Paradigm 8.0

Visual Paradigm para UML es una herramienta que acelera el desarrollo de aplicaciones, sirviendo de intermediario visual entre arquitectos, analistas y diseñadores de software, mediante un ambiente de modelado superior que posibilita un trabajo más fácil y dinámico. Esta herramienta profesional admite el ciclo de vida completo del software: análisis, diseño, construcción, pruebas y la implementación orientada a objetos (Softpedia, 2013). Algunas de sus características son:

- Soporta el lenguaje de modelado UML con una amplia documentación y todos sus diagramas de diseño.
- Diseño de diagrama de tarjetas CRC (clase-responsabilidad-colaborador).
- Es multiplataforma.
- Exporta diagramas a imágenes (PNG, JPG).

Se selecciona como herramienta para el modelado de las especificaciones de la aplicación el Visual Paradigm 8.0 teniendo en cuenta sus características y los beneficios que brinda para la construcción de software, especialmente referente al modelado. Permite su integración con varias bases de datos entre las que se encuentra PostgreSQL, lenguajes de programación como PHP y varios IDEs entre ellos NetBeans. No es gratuita, pero la UCI cuenta con una licencia para su uso, además cubre todos los flujos de trabajo que se requieren en el desarrollo del sistema en cuestión.

### 2.4 Herramientas IDE

Los Entorno Integrado de Desarrollo (IDEs -*Integrated Development Environment*) son un conjunto de herramientas para el programador, que suelen incluir en una misma suite, un buen editor de código, administrador de proyectos y archivos, enlace transparente a compiladores y *debugger* e integración con sistemas controladores de versiones o repositorios.

#### NetBeans 7.3

Es un IDE que permite a los programadores escribir, compilar, depurar y ejecutar programas. Contiene las herramientas para que los desarrolladores de *software* puedan crear aplicaciones desktop, enterprise, web, y aplicaciones móviles, con el lenguaje Java, así como también C/C++, PHP, JavaScript entre otros. Cuenta con un número importante de módulos entre los que se encuentra el módulo de desarrollo para aplicaciones web con PHP, el de Symfony2 que facilita el desarrollo de aplicaciones haciendo uso de este *framework*, y el módulo de subversión que desde el entorno permite manipular las versiones del código durante la implementación.

Se seleccionó NetBeans7.3 como IDE de desarrollo porque es un producto libre y gratuito sin restricciones de uso. Además ofrece soporte para Symfony2 y permite la integración de servidores, así como autocompletar y depurar el código de una manera muy sencilla mediante una consola de comandos.

### 2.5 Lenguaje de programación.

Es un lenguaje diseñado para describir el conjunto de sintaxis y reglas semánticas que debe ejecutar un determinado equipo. Es una técnica estándar de comunicación para entregarle instrucciones a un ordenador. Un lenguaje le da la capacidad al programador de especificarle al ordenador que tipo de datos actúan y que acciones tomar bajo una variada gama de circunstancias, utilizando un lenguaje relativamente próximo al lenguaje humano. (LProgramación, 2012)

#### 2.5.1 Lenguaje de programación del lado del servidor.

Los lenguajes del lado del servidor son aquellos que son reconocidos, ejecutados e interpretados por el propio servidor y que se envían al cliente en un formato comprensible.

#### PHP 5.3.8

Es un acrónimo recurrente que significa Procesador de Hipertexto (*Hypertext Pre-processor*), es un lenguaje de programación interpretado, usado para la creación de aplicaciones para servidores, o

creación de contenido dinámico para sitios Web. Su interpretación y ejecución se da en el servidor y el cliente sólo recibe el resultado de la ejecución. Ventajas que brinda:

- Lenguaje multiplataforma.
- Capacidad de conexión con la mayoría de los motores de bases de datos que se utilizan en la actualidad, destaca su conectividad con MySQL y PostgreSQL.
- Permite leer y manipular datos desde diversas fuentes, incluyendo datos que pueden ingresar los usuarios desde formularios HTML.
- Completamente orientado al desarrollo de aplicaciones web dinámicas con acceso a información almacenada en la base de datos.
- Es libre, por lo que se presenta como una alternativa de fácil acceso para todos.

### **2.5.2 Lenguaje de programación del lado del cliente.**

#### **Lenguaje de Marcado de Hipertexto (HTML 5.0)**

HTML es la abreviatura de *HyperText Markup Language* es un lenguaje del lado del cliente utilizado para presentar información en la *World Wide Web*. Su objetivo es describir cómo es una página de manera que examinando esa descripción el navegador del usuario final sea capaz de mostrarlo de la mejor manera posible (Ferrer,1998). HTML está basado en el uso de etiquetas, es sencillo, fácil de aprender, compatible con todos los sistemas operativos, permite cambiar y actualizar la información de manera rápida, así como crear páginas web con gran facilidad. Mediante su utilización se pueden incluir imágenes y ficheros multimedia en las páginas web, además de hiperenlaces que permiten acceder a otros documentos.

#### **Hoja de Estilo en Cascada (CSS)**

El gran auge de la Internet, así como el crecimiento del HTML para la creación de páginas web, provocaron un gran impulso de los lenguajes de hojas de estilos. A finales de 1996 el W3C (*World Wide Web Consortium*) publica la primera recomendación oficial del CSS (*Cascading Style Sheets* por su siglas en inglés), el cual es un lenguaje del lado del cliente definido como "(...) un lenguaje de hojas de estilos creado para controlar el aspecto o presentación de los documentos electrónicos definidos con HTML". CSS es imprescindible para crear páginas web complejas ya que es la mejor forma de separar los contenidos y su presentación, lo que permite un mejor entendimiento de estas, al mismo tiempo que facilita el trabajo de los desarrolladores (Eguíluz, 2009).

### JavaScript

Es un lenguaje de programación que se utiliza principalmente para crear páginas web dinámicas. Técnicamente es un lenguaje de programación interpretado, por lo que no es necesario compilar los programas para ejecutarlos. Se trata de un lenguaje de programación del lado del cliente, porque es el navegador el que soporta la carga de procesamiento. Está diseñado para controlar la apariencia y manipular los eventos dentro de la ventana del navegador Web y es soportado por la gran mayoría de los navegadores. Permite crear contenidos dinámicos y elementos de la página que tengan movimiento, cambien de color o cualquier otro dinamismo y permite ejecutar instrucciones como respuesta a las acciones del usuario, con lo que se pueden crear páginas interactivas con programas como: tablas de cálculo, calendarios y validar informaciones entradas por el usuario para verificar su veracidad o su forma de escribirse. (Eguíluz, 2008)

### 2.6 Framework

*“Un Framework en el desarrollo de software es una estructura de soporte definida, mediante la cual un proyecto puede ser organizado y desarrollado. Típicamente, puede incluir soporte de programas, bibliotecas o lenguaje interpretado entre otros software para ayudar a desarrollar y unir los diferentes componentes de un proyecto”* (Fabien Potencier, 2010) .Constituyen herramientas indispensables de alto valor para los desarrolladores, pues incluyen componentes o piezas pre-elaboradas de código que evitan la pérdida de tiempo y favorecen la seguridad y versatilidad de las aplicaciones.

#### Symfony 2.4.2

Symfony es un *framework* que simplifica el desarrollo de una aplicación web mediante la automatización de algunos de los patrones utilizados para resolver las tareas comunes. Separa la lógica del negocio, la lógica del servidor y la presentación de la aplicación web según el patrón Modelo Vista Controlador (Fabien Potencier, 2010). Proporciona estructura al código fuente, forzando al desarrollador a crear código más legible y más fácil de mantener. La capa de presentación utiliza plantillas y *layouts* que pueden ser creados por diseñadores HTML sin ningún tipo de conocimiento del framework. Los formularios incluyen validación automatizada y relleno automático de datos, lo que asegura la obtención de datos correctos y mejora la experiencia del usuario. En resumen Symfony2 es una framework maduro, rápido, configurable y ampliable que se adapta a casi cualquier arquitectura y requisitos del negocio.

Algunas de sus características son las siguientes (Eguíluz, 2013):

- Es fácil de instalar y configurar en la mayoría de las plataformas (con la garantía de que funciona correctamente en los sistemas Windows y Unix estándares).
- Es independiente del sistema gestor de bases de datos.
- Motor de plantillas twig.
- Es fácil de extender, lo que permite su integración con librerías desarrolladas por terceros.

Se selecciona el framework Symfony2 pues es un proyecto PHP de *software* libre que permite crear aplicaciones y sitios web rápidos y seguros de forma .es compatible con la mayoría de gestores de bases de datos y automatiza las tareas más comunes, permitiendo al desarrollador dedicarse por completo a los aspectos específicos de cada aplicación.

### **2.7 Sistema Gestor de Bases de Datos.**

Los sistemas de gestión de bases de datos (SGBD) son un tipo de *software* muy específico, dedicado a servir de interfaz entre la base de datos, el usuario y las aplicaciones que la utilizan. El propósito general de los sistemas de gestión de bases de datos es el de manejar de manera clara, sencilla y ordenada un conjunto de datos que posteriormente se convertirán en información relevante para una organización. Los SGBD proveen facilidades para la manipulación de grandes volúmenes de datos y disminuyen drásticamente los tiempos de desarrollo, aumentando la calidad del sistema desarrollado si son bien explotados por los desarrolladores. Los SGBD permiten definir los datos a distintos niveles de abstracción y manipular dichos datos, garantizando la seguridad e integridad de los mismos.

#### **PostgreSQL 9.1**

PostgreSQL es un sistema de gestión de bases de datos objeto-relacional, de propósito general, multiusuario y de código abierto, que soporta gran parte del estándar SQL y ofrece modernas características como consultas complejas, disparadores, vistas, integridad transaccional, control de concurrencia y que puede ser extendido por el usuario añadiendo tipos de datos, operadores, funciones agregadas, funciones ventanas y funciones recursivas, métodos de indexado y lenguajes procedurales. (PostgreSQL, 2010)

Las principales características que hacen de PostgreSQL el sistema de gestión de bases de datos de código abierto más avanzado del mundo son:



- Soporta consultas SQL declarativas, transacciones, optimización de consultas, herencia y arreglos.
- Soporta tipos de datos, operadores, funciones y métodos de acceso definidos por el usuario.
- Provee interfaces a diferentes lenguajes de programación como Object Pascal, Python, Perl, PHP, Java, Ruby, TCL, C/C++ y Pike.
- Soporta características objeto-relacionales (herencia, funciones y operadores polimórficos).
- Permite la gestión de diferentes usuarios, como también los permisos asignados a cada uno de ellos.

Se opta por la utilización de PostgreSQL pues este potente gestor de bases de datos, tiene prestaciones y funcionalidades equivalentes a muchos gestores de bases de datos comerciales como Oracle o MySQL. Posee diversas ventajas dentro de las cuales está su alto rendimiento, su bajo coste, su facilidad de configuración y aprendizaje, su accesibilidad a código fuente. Soporta potencia y flexibilidad adicional como son las restricciones (*constraints*), disparadores (*triggers*), reglas (*rules*) e integridad transaccional.

### **PgAdmin III**

Es una potente herramienta de administración de bases de datos está diseñada para responder a las necesidades de todos los usuarios, desde escribir simples consultas SQL hasta la elaboración de bases de datos complejas. Esta aplicación no requiere ningún controlador adicional para comunicarse con el servidor de base de datos, es desarrollado por una comunidad de especialistas de todo el mundo y está disponible en más de 30 idiomas (Ruiz, 2012). Es una aplicación libre, multiplataforma y diseñada para múltiples versiones de PostgreSQL y posee una amplia documentación.

## **2.8 Servidor Web**

Es un ordenador que usa el protocolo http para enviar páginas web al ordenador de un usuario cuando el usuario las solicita. Un servidor web es un programa que se ejecuta continuamente en un computador, manteniéndose a la espera de peticiones de ejecución que le hará un cliente o un usuario de Internet. El servidor web se encarga de contestar a las peticiones de un cliente o usuario de forma adecuada, entregando como resultado una página web o información de todo tipo de acuerdo a los comandos solicitados.

### **Apache 2.4**

Apache es un servidor Web, por su configurabilidad, robustez y estabilidad hacen que cada vez millones de servidores reiteren su confianza en este programa. Presenta entre otras características mensajes de error altamente configurables, bases de datos de autenticación y negociado de contenido, pero fue criticado por la falta de una interfaz gráfica que ayude en su configuración (CIBERAULA.2010). Algunas de las características de este servidor web son:

- Es una tecnología gratuita de código fuente abierto y multiplataforma.
- Trabaja con gran cantidad de lenguajes como Perl, PHP y otros lenguajes de script.
- Soporte de autenticación HTTP.

### **2.9 Conclusiones parciales.**

En el presente capítulo se selecciona como metodología de desarrollo XP la cual proporciona una guía para ordenar las actividades y desarrollar los artefactos, lo que facilita la organización del trabajo, contribuyendo a la obtención de un sistema en el tiempo establecido. Se definieron los elementos que componen el entorno tecnológico para la construcción del sistema: Postgres9.1, Netbeans7.3, Symfony2, Apache 2.4 y Visual Paradigm8.0; los cuales son adecuados para el proceso de desarrollo ágil como complemento de la metodología seleccionada.

# CAPÍTULO 3

## Análisis y diseño de la propuesta de solución.

---

### Introducción

En este capítulo se presenta el Sistema para automatizar la gestión de la información científico técnica del centro GEYSED como solución a la problemática de la investigación. Se definen las Historias de Usuario (HU) en las cuales están recogidos los requisitos funcionales del sistema, así como los patrones de diseño y se representan las clases del sistema mediante las Tarjetas Clase-Responsabilidad-Colaborador (CRC). Se propone un modelo de dominio y un diagrama de clases del diseño que se utiliza como complemento de la metodología definida, para lograr un mejor entendimiento del *software*.

### 3.1 Descripción del sistema.

Para darle solución a la situación problemática, se propone desarrollar un sistema que logre centralizar la información relacionada con el desempeño del centro GEYSED en cuanto ciencia y técnica se refiere. La aplicación facilita la gestión de un conjunto de indicadores, los cuales permiten visualizar el desempeño del centro en las siguientes actividades: publicaciones, premios, composición del claustro, capacitación, patentes y registros, proyectos I+D y resultados introducidos. El sistema a desarrollar es una aplicación web, denominada Sistema para la gestión de la actividad científica (SIGAC), con el cual se pueden realizar las siguientes operaciones:

- Gestionar la información asociada a la composición del claustro de profesionales del centro.
- Gestionar las actividades realizadas en cuanto a investigación y superación, entiéndase: publicaciones, premios obtenidos, trabajos presentados en eventos y capacitación recibida u ofertada.
- Gestionar las actividades que tributan a la CTI que son registradas a nivel de centro como son: las patentes y registros, los proyectos I+D y los resultados introducidos (productos terminados).
- Representación gráfica del progreso de la actividad investigativa y de superación del centro.
- Exportar a formato PDF la información registrada en el sistema referente a las actividades científicas y de postgrado.

### 3.2 Modelo de Dominio.

La metodología XP no especifica una técnica para definir el negocio, pero este proceso es necesario realizarlo de manera entendible para luego poder realizar la implementación. La flexibilidad que posee XP permite incluir dentro del proceso de desarrollo de *software* artefactos que son generados en otras metodologías. Una de las técnicas más utilizadas para lograr el entendimiento del sistema que se desea desarrollar es el modelo de dominio. El cual es una representación visual estática del entorno, un diagrama con los objetos que existen (reales) relacionados con el proyecto, ayuda a comprender mejor los elementos involucrados en el negocio. En la Figura 1 se muestra el modelo de dominio de la propuesta de solución.

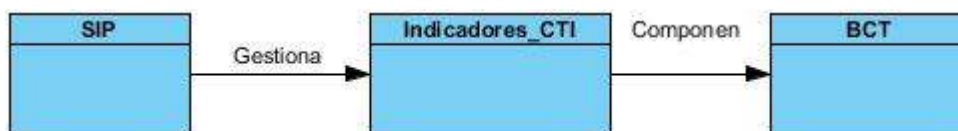


Figura 1: Modelo de Dominio

**SIP:** Subdirector de Investigación es el encargado de la gestión de la información asociada a las investigaciones y postgrados en el centro GEYSED

**Indicadores\_CTI:** Indicadores de ciencia, tecnología e innovación que agrupan las actividades de investigación y postgrado de los profesionales y del centro: publicaciones realizadas, permios obtenidos, trabajos presentados en eventos, capacitación ofertada o recibida, proyectos I+D, patentes y registros y resultados introducidos

**BCT:** Es el Balance de Ciencia y Técnica el cual constituye un resumen del cumplimiento de los Indicadores\_CTI en un período de tiempo determinado, tomando como referencia el Plan de Indicadores que se define para cada año fiscal .

El SIP confecciona un Plan de Indicadores\_CTI para el centro, para un período que comprende un año fiscal. Además debe recopilar la información asociada al cumplimiento de la planificación realizada, a partir de la cual se elaboran estadísticas que constituyen el BCT.

### 3.3 Lista de Reserva del Producto(LRP).

La Lista de Reserva del Producto es una tabla que contiene los requisitos funcionales que debe cumplir la aplicación que se desea realizar. Estos son ordenados según la prioridad para su implementación, ubicados en Muy Alta, Alta, Media y Baja, en la categoría Baja aparecen los requisitos de menor complejidad; además de los requisitos no funcionales del sistema. La estimación, es el

## CAPÍTULO 3: Análisis y diseño de la propuesta de solución.

esfuerzo que cuesta la implementación de cada Historia de Usuario (HU) y asociadas a estas sus entregas. El criterio utilizado para estimar el desempeño del proyecto en la presente investigación es la medida de puntos; un punto se considera como una semana ideal de trabajo donde los miembros de los equipos de desarrollo trabajan el tiempo planeado sin ningún tipo de interrupción.

A continuación se muestra en la Tabla 1 la Lista de Reserva del Producto del sistema a implementar, donde se relacionan los 15 requisitos funcionales identificados, de ellos 8 con prioridad muy alta, 3 con prioridad alta y 4 con prioridad media. Además se detallan los requerimientos no funcionales que debe cumplir el sistema para su correcto funcionamiento.

**Tabla 1:Lista de Reserva del Producto**

Ítem *	Descripción	Estimación	Estimado por
<b>Prioridad :Muy Alta</b>			
1	Gestionar Claustro	0.5	Analista.
2	Gestionar publicaciones de un profesional.	0.5	Analista.
3	Gestionar premios de un profesional.	0.5	Analista.
4	Gestionar capacitación de un profesional.	0.5	Analista.
5	Gestionar trabajos de un profesional.	0.5	Analista.
6	Gestionar proyectos I+D del centro.	0.5	Analista.
7	Gestionar patentes y registros del centro.	0.5	Analista.
8	Gestionar resultados introducidos en el centro.	0.5	Analista.
<b>Prioridad: Alta</b>			
9	Gestionar Categorías.	0.5	Analista.
10	Exportar a PDF.	0.5	Analista.
11	Graficar indicadores.	1	Analista
<b>Prioridad :Media</b>			
12	Gestionar usuarios del sistema	0.5	Analista.
13	Autenticar usuario.	0.5	Analista.
14	Gestionar Departamento.	0.5	Analista.
15	Gestionar Centro.	0.5	Analista.
<b>Prioridad :Baja</b>			
<b>Portabilidad.</b>			
<ul style="list-style-type: none"> <li>➤ El sistema debe ser multiplataforma, permitiendo disponer del mismo en cualquier sistema operativo (Microsoft Windows 2000/NT+ o Distribución de GNU/Linux).</li> </ul>			
<b>Interfaz</b>			

## CAPÍTULO 3: Análisis y diseño de la propuesta de solución.

<ul style="list-style-type: none"><li>➤ El sistema debe presentar una interfaz amigable que permita una fácil interacción con el mismo así como un rápido acceso a la información buscada, por ejemplo botones con íconos sugerentes y alternativa textual.</li></ul>
<b>Requisitos de hardware</b> <u>PC Servidor</u> <ul style="list-style-type: none"><li>➤ Capacidad mínima de memoria RAM de 1 GB, recomendable 2 GB.</li><li>➤ Velocidad de procesamiento del microprocesador 1GHz o superior.</li><li>➤ Capacidad de almacenamiento adecuada, recomendable 30 GB libres (o superior).</li></ul> <u>PC Cliente</u> <ul style="list-style-type: none"><li>➤ Memoria RAM 512 MB, recomendable 1 GB (o superior).</li><li>➤ Velocidad de procesamiento del microprocesador 1GHz o superior.</li></ul>
<b>Requisitos de software</b> <u>PC Servidor</u> <ul style="list-style-type: none"><li>➤ Instalación del servidor web Apache 2.4.4.</li><li>➤ Instalación del servidor de base de datos PostgreSQL 9.1.</li><li>➤ Instalación de PHP 5.3.8.</li><li>➤ Sistema operativo: Microsoft Windows 2000/NT+ o Distribución de GNU/Linux.</li></ul> <u>PC Cliente</u> <ul style="list-style-type: none"><li>➤ Navegador web de internet con soporte para HTML 5 y CSS 3, Recomendable Firefox 10+ o Google Chrome 14+.</li><li>➤ Sistema operativo: Microsoft Windows 2000/NT+ o Distribución de GNU/Linux.</li></ul>
<b>Usabilidad</b> <ul style="list-style-type: none"><li>➤ Emplear perfiles de usuario: diferenciar las interfaces y opciones para los usuarios que accedan al sistema según los diferentes roles que estos tengan dentro del sistema.</li></ul>

### 3.4 Historias de Usuario

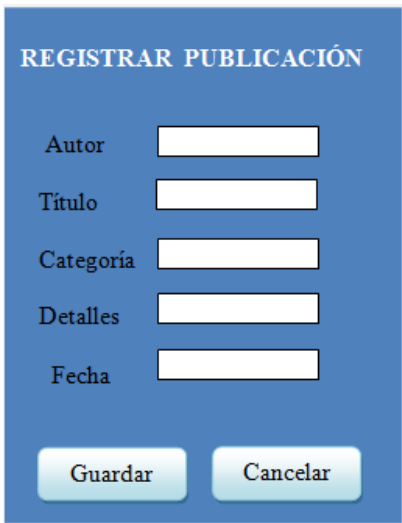
Son la técnica utilizada para especificar los requisitos del *software* se trata de tarjetas en las cuales el cliente describe brevemente las características que el sistema debe poseer, sean requisitos funcionales o no funcionales. Las HU son artefactos que constan de tres o cuatro líneas escritas en un lenguaje no técnico, en las cuales no se debe hablar de posibles algoritmos para su implementación o de diseños de base de datos adecuados.

Son usadas para estimar tiempos de desarrollo de la parte de la aplicación que describen y se utilizan en la fase de pruebas, para verificar si el programa cumple con lo que se especifica en las Historias de Usuarios. Para el presente trabajo se obtuvieron un total de 15 HU que se desarrollan en 3 iteraciones.

## CAPÍTULO 3: Análisis y diseño de la propuesta de solución.

A continuación se muestra en la Tabla 2 la HU Gestionar publicaciones de un profesional, el resto pueden ser consultadas en el Anexo II.

Tabla 2: HU\_2 Gestionar publicaciones de un profesional.

Historia de Usuario	
Número: HU_2	Nombre Historia de Usuario: Gestionar publicaciones de un profesional
Modificación de Historia de Usuario Número: Ninguna	
Usuario: Gabriel D. Pérez Fernández	Iteración Asignada: 1
Prioridad en Negocio: Alta	Puntos Estimados: 1 semana
Riesgo en Desarrollo: Alto	Puntos Reales: 1 semana
<b>Descripción:</b> la presente HU tiene como objetivo permitir adicionar, modificar, eliminar y visualizar las publicaciones realizadas por un profesional. Se incluyen los datos básicos del profesional como el nombre, apellidos; además de los datos necesarios para el registro de una publicación. De un publicación se debe conocer: categoría (libro, <i>Web of Science</i> , Nivel1_2_3_4), título, autor, fecha, detalles (indexación o ISSN, URL).	
<b>Observaciones:</b> El usuario debe completar los datos que se le solicitan y el sistema valida la correcta entrada de los mismos.	
<b>Prototipo de interface:</b> 	

### 3.5 Plan de Iteraciones

Después de contar con una definición de las historias de usuarios se confecciona el plan de iteraciones, con el objetivo de tener una planificación del trabajo. En este artefacto es donde los desarrolladores y el cliente establecen los tiempos de implementación de las historias de usuarios y la prioridad con que serán desarrolladas. Se identificaron 15 Historias de Usuario planificándose su implementación en 3 iteraciones comenzando por las HU de mayor prioridad del negocio. A continuación en la Tabla 3 se muestra el Plan de Iteraciones del sistema a implementar.

Tabla 3: Plan de Iteración del sistema

Iteración	Descripción de la iteración	Orden de la HU a implementar	Duración total
1	En esta iteración se implementan las HU con prioridad Muy Alta. Estas están asociadas a la gestión de las actividades científicas de los profesionales (publicaciones, premios, trabajos y capacitación) y actividades CTI específicas del centro (patentes y registros, proyectos I+D y resultados introducidos).	1-2-3-4-5-6-7-8	4 semanas
2	En esta iteración se desarrollan las HU con prioridad Alta las cuales están asociadas a la gestión de categorías, exportar a PDF y graficar indicadores. Se corrigen errores o disconformidades con las HU implementadas en la iteración anterior.	9-10-11	2 semanas
3	En esta iteración se implementan las HU con prioridad Media y Baja; estas están asociadas a la gestión de usuarios del sistema, a la autenticación del sistema, gestión de los departamentos y centros. Se corrigen errores o disconformidades con las HU implementadas en la iteración anterior.	12-13-14-15	2 semanas

### 3.6 Arquitectura.

Según el estándar de la IEEE (*Institute of Electrical and Electronics Engineers*): “La Arquitectura de Software es la organización fundamental de un sistema encarnada en sus componentes, las relaciones entre ellos y el ambiente y los principios que orientan su diseño y evolución “. De modo general una arquitectura de *software* permite analizar la efectividad del diseño para cumplir con los requisitos establecidos.



### Arquitectura Cliente-Servidor

La arquitectura cliente-servidor es un modelo de aplicación distribuida en el que las tareas se reparten entre los proveedores de recursos o servicios, llamados servidores, y los demandantes, llamados clientes. Un cliente realiza peticiones a otro programa, el servidor, que le da respuesta. En esta arquitectura la capacidad de proceso está repartida entre los clientes y los servidores, aunque son más importantes las ventajas de tipo organizativo debido a la centralización de la gestión de la información y la separación de responsabilidades, lo que facilita y clarifica el diseño del sistema.

**Cliente:** el cliente es el proceso que permite al usuario formular los requisitos y pasarlos al servidor. El cliente normalmente maneja todas las funciones relacionadas con la manipulación y despliegue de datos, por lo que están desarrollados sobre plataformas que permiten construir interfaces gráficas de usuario, además de acceder a los servicios distribuidos en cualquier parte de una red. Las funciones que lleva a cabo el proceso cliente se resumen en los siguientes puntos:

- Administrar la interfaz de usuario.
- Interactuar con el usuario.
- Procesar la lógica de la aplicación y hacer validaciones locales.
- Recibir resultados del servidor.

**Servidor:** es el proceso encargado de atender a múltiples clientes que hacen peticiones de algún recurso administrado por él. El servidor normalmente maneja todas las funciones relacionadas con la mayoría de las reglas del negocio y los recursos de datos. Las funciones que lleva a cabo el proceso servidor se resumen en los siguientes puntos:

- Aceptar los requisitos de bases de datos que hacen los clientes.
- Procesar requisitos de bases de datos.
- Procesar la lógica de la aplicación y realizar validaciones a nivel de bases de datos.

La ventaja más importante de este modelo es que es una arquitectura distribuida. Se puede hacer un uso efectivo de los sistemas en red con muchos procesadores distribuidos. Es fácil añadir un nuevo servidor e integrarlo con el resto del sistema o actualizar los servidores de forma transparente sin

afectar al resto del sistema. En la Figura 2 se muestra una representación de la arquitectura cliente servidor.



Figura 2: Arquitectura Cliente-Servidor.

### 3.7 Patrón de Arquitectura

El patrón arquitectónico es el nivel en el cual la arquitectura de *software* define la estructura básica de un sistema, pudiendo estar relacionado con otros patrones y representa una plantilla de construcción que provee un conjunto de subsistemas aportando las normas para su organización (Bahit, 2011). Uno de los patrones más utilizados en las aplicaciones web, y que además es el empleado para el desarrollo de la solución propuesta, es el patrón **Modelo-Vista-Controlador**.

- **Modelo:** La capa del modelo define la lógica de negocio se encarga de acceder a los datos, actuando como intermediario entre el controlador y la base de datos.
- **Vista:** es la encargada de mostrar la información al usuario de forma gráfica permitiendo la interacción con la aplicación.
- **Controlador:** es el intermediario entre la vista y el modelo, controla las interacciones del usuario, solicitando los datos al modelo y entregándolos a la vista para que esta los presente.

A continuación para una mejor comprensión del patrón se muestra la Figura 3:

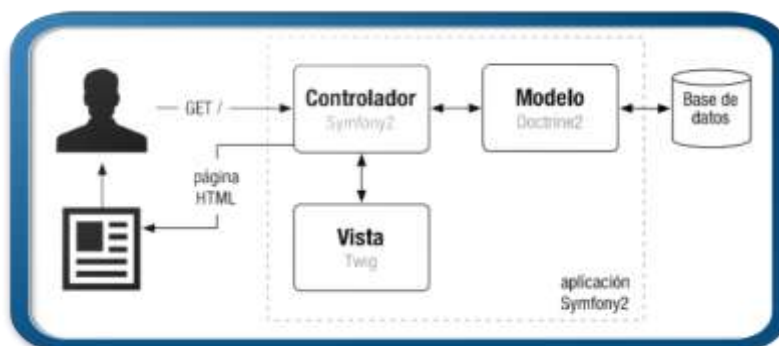


Figura 3: Patrón Modelo-Vista-Controlador (Potencier, 2008).

### 3.8 Patrones de Diseño.

Los patrones de diseño definen cómo construir un *software*, cómo utilizar las clases y los objetos de forma conocida. Un patrón de diseño es una solución a un problema de diseño. Para que una solución sea considerada un patrón debe poseer ciertas características. Una de ellas es que debe haber comprobado su efectividad resolviendo problemas similares en ocasiones anteriores. Otra es que debe ser reusable, lo que significa que es aplicable a diferentes problemas de diseño en distintas circunstancias (Larman, 1999).

#### **Los patrones GRASP**

Los patrones GRASP describen principios fundamentales de la asignación general de responsabilidades dentro de la programación orientada a objetos mediante la aplicación de “buenas prácticas” recomendables en el diseño de *software*. Para el desarrollo del sistema se usaron los siguientes patrones:

**Patrón Experto:** Este patrón garantiza que la responsabilidad de la creación de un objeto o la implementación de un método, recaiga sobre la clase que conoce toda la información necesaria para crearlo lo que contribuye a un adecuado encapsulamiento, favoreciendo la robustez y fácil mantenimiento del sistema.

El patrón se evidencia mediante el empleo del ORM *Doctrine*, *Symfony2* el cual crea entidades para manipular la información de la base de datos. Estas entidades contienen la información necesaria de la tabla que representan, al mismo tiempo que cuentan con las funcionalidades necesarias para el trabajo con los datos.

**Patrón Creador:** Se asigna la responsabilidad a una clase de crear cuando contiene, agrega, compone, almacena o usa otra clase, lo que brinda una alta posibilidad de reutilizar la clase creadora.

Su empleo se evidencia mediante el uso del contenedor de servicios de *Symfony2*, el cual es un objeto PHP que gestiona la creación de instancias de objetos. Para la utilización del mismo, solo es necesario añadir una configuración al archivo **config.yml** del *framework*, especificándole el nombre del servicio, así como el **namespace** de las clases PHP. El contenedor que incluye *Symfony2* se encarga de procesar esa configuración y preparar todas las dependencias entre las clases.

**Controlador:** El patrón controlador ayuda a identificar quién debe ser el responsable de la creación (o instanciación) de nuevos objetos o clases. Su empleo se evidencia mediante la utilización de un controlador frontal (*app.php*), es decir, un solo archivo PHP que se encarga de tratar todas las peticiones que llegan a la aplicación. Mediante su uso cada petición se maneja de igual manera, pues

## CAPÍTULO 3: Análisis y diseño de la propuesta de solución.

---

el enrutado de diferentes URLs a diferentes partes de la aplicación se realiza internamente, en lugar de URLs individuales ejecutando diferentes archivos PHP.

**Patrón Bajo Acoplamiento:** El bajo acoplamiento está dado por disminuir al máximo la interdependencia entre las clases y el poco impacto al introducirse cambios en los objetos. Este patrón se evidencia mediante la inyección de dependencias que provee Symfony2. La inyección de dependencias consiste en pasar a las clases todos los objetos que necesitan ya creados y configurados, logrando así reducir las dependencias entre las clases de la aplicación en la creación de las clases.

**Alta Cohesión:** La alta cohesión está dada por la relación inversa que existe entre la responsabilidad que recae sobre una clase o subsistema, la complejidad y cantidad de trabajo que realiza. Su utilización se hace evidente ya que las clases de la aplicación tienen una función bien definida dentro del sistema, por lo que cada una de ellas es responsable de las tareas relacionadas con esta.

### **Los patrones GOF:**

Los patrones GoF (*Gang of Four*), describen las formas comunes en que diferentes tipos de objetos pueden ser organizados para trabajar unos con otros. Tratan la relación entre clases, la combinación de clases y la formación de estructuras de mayor complejidad. Permiten crear grupos de objetos que ayudan a realizar tareas complejas. Estos patrones pueden ser de tres tipos: de creación, estructurales y de comportamiento.

**Singleton:** Permite garantizar que una clase sólo tiene una única instancia, proporcionando un punto de acceso global a la misma. Este patrón permite que el acceso a una instancia esté más controlado, se reduce el espacio de nombres, permite refinamientos en las operaciones y en la representación. Este patrón se evidencia mediante el uso de los namespace, permitiendo que dos o más clases de una misma aplicación puedan compartir su nombre. El único requisito es que sus namespaces sean diferentes, de forma que la aplicación sepa en todo momento cuál se está utilizando.

**Decorator (Decorador):** Es un patrón estructural que sirve para diseñar las interconexiones entre los objetos, a la vez que estudia cómo estos se relacionan en tiempos de ejecución. Su utilización se evidencia en el desarrollo de la propuesta de solución debido a la existencia de los archivos `base.html.twig` y `backend.html.twig`; estos almacenan el código HTML común a todas las páginas de la aplicación, evitando así tener que repetirlo en cada una de ellas.

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

La metodología XP propone el uso de las tarjetas Clase-Responsabilidad-Colaborador (CRC) con el objetivo de desarrollar una representación organizada de las clases. Esta técnica consiste en la confección de tarjetas índices estándar que representan clases, la clase a la que pertenece el objeto se puede escribir en la parte de arriba de la tarjeta, en una columna a la izquierda se escriben las responsabilidades u objetivos que debe cumplir el objeto y a la derecha, las clases que colaboran con cada responsabilidad.

- **Clase:** Cualquier persona, cosa, evento, concepto, pantalla o reporte. Además consiste en crear una tarjeta para cada clase.
- **Responsabilidades:** Son todos los servicios que proporciona para todos los contratos que soporta la lista de servicios que una instancia de una clase puede pedir a una instancia de otra.
- **Colaboraciones:** Aquellas clases con las que trabaja en conjunto para llevar a cabo sus responsabilidades.

A continuación se muestra en la Tabla.4 la tarjeta CRC asociada a la clase Profesional el resto pueden ser consultadas en el Anexo III.

Tabla 4: Tarjeta CRC Clase Profesional

Tarjeta CRC	
Profesional.php	
Responsabilidades	Colaboradores
Se implementan los métodos para la creación, modificación, visualización y eliminación de un profesional.	ProfesionalController.php
Contiene el constructor del formulario que permite la inserción de los datos para la posterior creación, modificación y visualización de la información de un Profesional.	ProfesionalType.php

## 3.10 Diagrama de Clases

La metodología XP no incluye entre sus artefactos el diagrama de clases, pero en la solución propuesta se realiza el mismo para lograr de una forma más sencilla y organizada el trabajo. Los diagramas de clases se utilizan para representar la estructura estática de un sistema incluyendo una colección de elementos tales como clases y relaciones. Un diagrama de clases del diseño describe gráficamente las especificaciones de las clases del *software* y de las interfaces en una aplicación. En la Figura 4 se muestra el diagrama de clases del diseño asociado a la HU\_2 Gestionar publicaciones de un profesional, el resto de los diagramas pueden ser consultados en el Anexo IV.

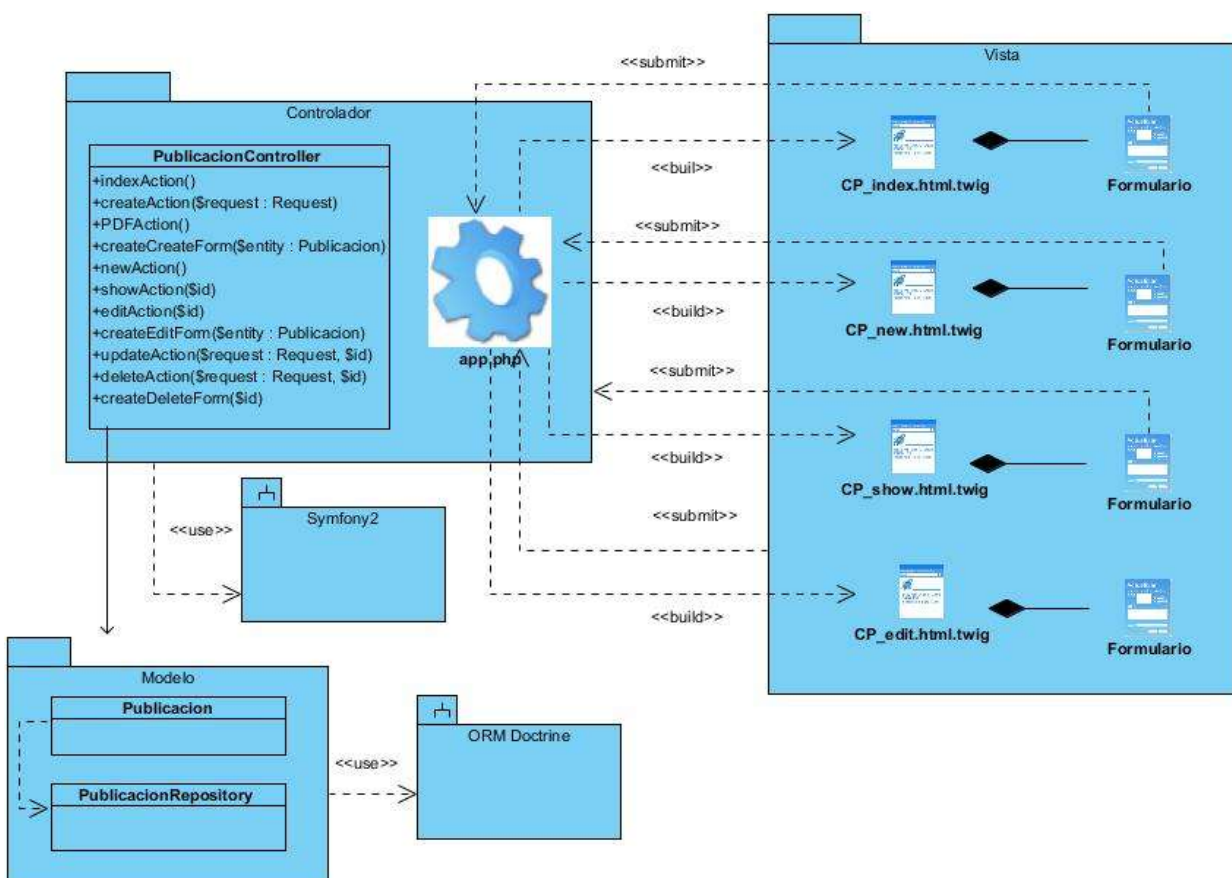


Figura 4: Diagrama de clases del diseño asociado a la HU\_2 Gestionar publicaciones

El diagrama representado en la Figura 4 las clases están organizadas por paquetes siguiendo la estructura del patrón arquitectónico Modelo-Vista-Controlador. Las clases de interfaz se encuentran en el paquete Vista, las clases de lógica del negocio en el paquete Controlador y las clases de acceso a datos en el paquete Modelo. Además se representan los subsistemas con los cuales interactúa el sistema: Symfony2 y ORM Doctrine.

## Modelo de Datos

Un modelo de datos es un conjunto de conceptos que sirven para describir la estructura de una base de datos: los datos, las relaciones entre ellos y las restricciones que deben cumplirse sobre los mismos. En la Figura 5 se representa el modelo físico de datos de la propuesta de solución.

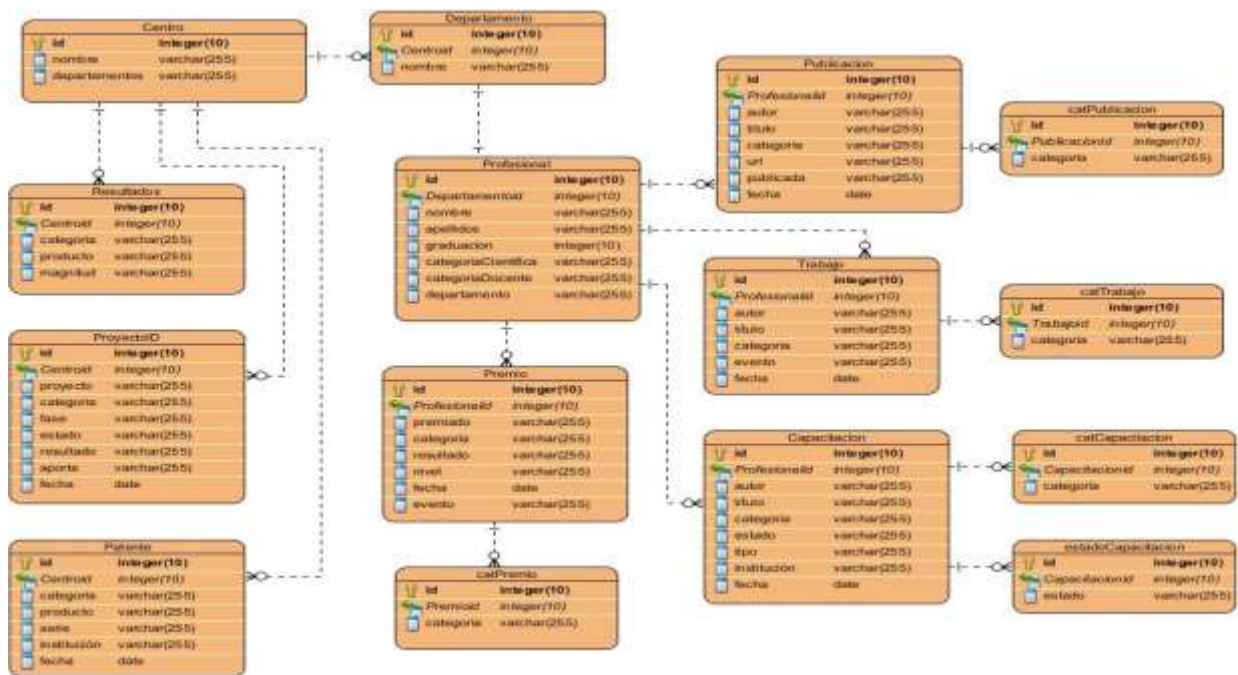


Figura 5: Modelo de datos del sistema.

## 3.11 Conclusiones parciales

En el presente capítulo se realiza el análisis y diseño de la propuesta de solución, obteniéndose una mejor visión de lo que el cliente desea. Se obtiene un mejor entendimiento de la aplicación a través de la propuesta del sistema y el modelo de dominio descrito; la confección de las HU facilita la definición de las funcionalidades del sistema. Se describe la planificación del tiempo de desarrollo de las iteraciones a ejecutar y el orden en que se implementarán las HU. La elaboración de las tarjetas CRC facilita la representación organizada de las clases más relevantes del sistema y sus relaciones.



# CAPÍTULO 4

## Implementación y pruebas.

### Introducción

En este capítulo se describe la fase de implementación y prueba del sistema, se detallan las tareas de la ingeniería generadas por las correspondientes HU definidas. Se muestra el resultado de las pruebas realizadas al *software* con el propósito de garantizar que fue adecuada la implementación del mismo.

### 4.1 Tareas de la Ingeniería

La metodología XP propone que la implementación debe ejecutarse de forma iterativa e incremental, alcanzando al final de cada iteración un producto funcional que debe ser examinado y mostrado al cliente, de esta forma se garantiza una constante retroalimentación entre los desarrolladores y clientes. El equipo de desarrollo evalúa cada HU y las divide en tareas las cuales son asignadas a los programadores para ser implementadas durante la iteración correspondiente, donde cada una de estas representa una característica del sistema. A continuación en las Tablas 5-6-7-8, se muestran las tareas de la ingeniería asociadas a la HU\_2 Gestionar publicaciones de un profesional, el resto de las tareas de la ingeniería asociadas a las HU pueden ser consultadas en el Anexo V.

**Tabla 5: Tarea N°1: Adicionar publicaciones de un profesional.**

Tarea de Ingeniería	
<b>Número Tarea:</b> 1	<b>Número Historia de Usuario:</b> HU_2 Gestionar publicaciones de un profesional.
<b>Nombre Tarea:</b> Adicionar publicaciones de un profesional	
<b>Tipo de Tarea :</b> Desarrollo	<b>Puntos Estimados:</b> 0.5
<b>Fecha Inicio:</b>	<b>Fecha Fin:</b>
<b>Programador Responsable:</b> Gabriel D. Pérez Fernández	
<b>Descripción:</b> La presente tarea de ingeniería tiene como objetivo permitir adicionar los datos a una publicación realizada por un profesional. Se implementan métodos en las clases Publicacion.php,	



PublicacionType y PublicacionController.php las cuales garantiza la creación de un tabla en la BD para las publicaciones y la construcción del formulario (buildForm) y las acciones asociadas a la inserción de un publicación (createAction, createCreateForm). Además se define la ruta de acceso a la página correspondiente a la gestión de publicaciones.

**Tabla 6: Tarea N°2: Modificar publicaciones de un profesional.**

Tarea de Ingeniería	
<b>Número Tarea:</b> 2	<b>Número Historia de Usuario:</b> HU_2 Gestionar publicaciones de un profesional.
<b>Nombre Tarea:</b> Modificar publicaciones de un profesional	
<b>Tipo de Tarea :</b> Desarrollo	<b>Puntos Estimados:</b> 0.5
<b>Fecha Inicio:</b>	<b>Fecha Fin:</b>
<b>Programador Responsable:</b> Gabriel D. Pérez Fernández	
<p><b>Descripción:</b> La presente tarea de ingeniería tiene como objetivo permitir modificar los datos a una publicación realizada por un profesional. Se implementan métodos en las clases Publicacion.php, PublicacionType y PublicacionController.php las cuales garantiza la creación de un tabla en la BD para las publicaciones y la construcción del formulario (buildForm) y las acciones asociadas a la inserción de un publicación (editAction, createEditForm). Además se define la ruta de acceso a la página correspondiente a la gestión de publicaciones.</p>	

**Tabla 7: Tarea N°3: Eliminar publicaciones de un profesional.**

Tarea de Ingeniería	
<b>Número Tarea:</b> 3	<b>Número Historia de Usuario:</b> HU_2 Gestionar publicaciones de un profesional.
<b>Nombre Tarea:</b> Eliminar publicaciones de un profesional	
<b>Tipo de Tarea :</b> Desarrollo	<b>Puntos Estimados:</b> 0.5
<b>Fecha Inicio:</b>	<b>Fecha Fin:</b>
<b>Programador Responsable:</b> Gabriel D. Pérez Fernández	
<p><b>Descripción:</b> La presente tarea de ingeniería tiene como objetivo permitir modificar los datos a una publicación realizada por un profesional. Se implementan métodos en las clases Publicacion.php, PublicacionType y PublicacionController.php las cuales garantiza la creación de un tabla en la BD</p>	

para las publicaciones y la construcción del formulario (buildForm) y las acciones asociadas a la inserción de un publicación (deleteAction, createDeleteForm). Además se define la ruta de acceso a la página correspondiente a la gestión de publicaciones.

**Tabla 8: Tarea Nº4: Visualizar publicaciones de un profesional.**

Tarea de Ingeniería	
<b>Número Tarea:</b> 3	<b>Número Historia de Usuario:</b> HU_ 2 Gestionar publicaciones de un profesional.
<b>Nombre Tarea:</b> Visualizar publicaciones de un profesional	
<b>Tipo de Tarea :</b> Desarrollo	<b>Puntos Estimados:</b> 0.5
<b>Fecha Inicio:</b>	<b>Fecha Fin:</b>
<b>Programador Responsable:</b> Gabriel D. Pérez Fernández	
<p><b>Descripción:</b> La presente tarea de ingeniería tiene como objetivo permitir visualizar los datos de las publicaciones realizada por un profesional. Se implementan métodos en las clases Publicacion.php, PublicacionType y PublicacionController.php las cuales garantiza la creación de un tabla en la BD para las publicaciones y la construcción del formulario (buildForm) y las acciones asociadas a la inserción de un publicación (showAction). Además se define la ruta de acceso a la página correspondiente a la gestión de publicaciones.</p>	

## 4.2 Pruebas

La prueba de software es un elemento clave para conocer el adecuado funcionamiento de un sistema informático, presentando como objetivos: detectar defectos en el software, comprobar que los requisitos se han implementado correctamente, identificar y asegurar que los defectos encontrados se han corregido antes de entregar el software al cliente (Acuña, 2010). En el presente acápite se describe el método que se selecciona para realizar las pruebas, la técnica que se utiliza para diseñar los casos de pruebas y los resultados obtenidos .

### 4.2.1 Prueba de Caja Negra.

Para la realización de las pruebas al sistema se utiliza el método de **Caja Negra**, el cual se centra en los requisitos funcionales del software. Se realiza sobre la interfaz del *software* mediante la confección de los casos de prueba basados en las HU definidas para el sistema. Para desarrollar las prueba de caja negra existen varias técnicas entre ella la de **Partición de Equivalencia**, la cual divide el campo

de entrada de entrada de un programa en clases de datos a partir de las cuales pueden derivarse casos de prueba. Es una de las técnicas más efectivas pues permite examinar los valores válidos e inválidos de las entradas existentes en el *software*. El objetivo del diseño de casos de prueba es que sean efectivos descubriendo defectos en los programas y muestre que el sistema satisface los requisitos bajo los cuales fue desarrollado.

Se definieron 15 casos de pruebas correspondientes a las 15 HU definidas para el sistema, los cuales se realizaron con el objetivo de determinar que las funcionalidades han sido implementadas adecuadamente logrando satisfacer las necesidades del cliente y demostrar que sus funciones son operativas. A continuación se muestran en las Tablas 9-10-11 los elementos que componen el caso de prueba definido para la HU\_2 Gestionar publicación de un profesional, el resto de los casos de prueba pueden ser consultados en el Anexo VI.

**Tabla 9: Caso de Prueba: Gestionar publicación (SC\_1: Adicionar Publicación)**

Nombre de la sección	Escenarios de la sección	Descripción de la Funcionalidad	Flujo central
SC_1:Adicionar publicación	EC_1.1: Adicionar publicación con éxito	<p>El sistema muestra una interfaz donde el usuario añade los datos de la nueva publicación:</p> <ul style="list-style-type: none"> <li>-autor</li> <li>-título</li> <li>-categoría</li> <li>-detalles</li> <li>-fecha</li> </ul> <p>El usuario entra los datos asociados a la publicación que se desea añadir y hace clic izquierdo en el botón guardar. El sistema guarda los cambios, visualiza los datos de la nueva publicación.</p>	<p>SIGAC “Módulo Administración”</p> <ol style="list-style-type: none"> <li>1-Hacer clic izquierdo en el menú Indicadores</li> <li>2-Hacer clic izquierdo en la opción Publicación</li> <li>3- Hacer clic izquierdo en el botón Registrar Publicación.</li> <li>4-El sistema muestra la interfaz para Registrar una Publicación</li> <li>5-El usuario completa los campos especificados</li> <li>6-Hacer clic izquierdo en el botón Guardar</li> <li>7-El sistema muestra los datos de la publicación registrada.</li> <li>8-Hacer clic izquierdo en el botón Atrás para regresa a la interfaz de administración.</li> </ol>

## CAPÍTULO 4: Implementación y pruebas.

	<p>EC_1.2: Añadir publicación sin éxito.</p>	<p>El sistema muestra un mensaje de información "Debe llenar este campo" en los campos que nos fueron completados. El usuario hace clic izquierdo en el botón Aceptar para cerrar la ventana que contiene el mensaje de error.</p>	<p>SIGAC "Módulo Administración"</p> <ol style="list-style-type: none"> <li>1-Hacer clic izquierdo en el menú Indicadores</li> <li>2-Hacer clic izquierdo en la opción Publicación</li> <li>3- Hacer clic izquierdo en el botón Registrar Publicación.</li> <li>4-El sistema muestra la interfaz para Registrar una Publicación</li> <li>5-El usuario completa los campos especificados</li> <li>6-Hacer clic izquierdo en el botón Guardar</li> <li>7-Se muestra un mensaje en los campos que no fueron completados.</li> <li>8-Rellenar los campos que no fueron completados.</li> <li>9-Hacer clic izquierdo en el botón Guardar</li> <li>10-El sistema muestra los datos de la publicación registrada.</li> <li>11-Hacer clic izquierdo en el botón Atrás para regresa a la interfaz de administración.</li> </ol>
	<p>EC_1.3: Cancelar petición.</p>	<p>El usuario hace clic izquierdo en el botón Cancelar. El sistema no guarda los cambios y regresa a la interfaz de administración.</p>	<p>SIGAC "Módulo Administración"</p> <ol style="list-style-type: none"> <li>1-Hacer clic izquierdo en el menú Indicadores</li> <li>2-Hacer clic izquierdo en la opción Publicación</li> <li>3- Hacer clic izquierdo en el botón Registrar Publicación.</li> <li>4-El sistema muestra la interfaz para Registrar una Publicación</li> <li>5- Hacer clic izquierdo en el botón Cancelar y el sistema regresa a la interfaz de administración</li> </ol>

	EC_1.4: Añadir publicación sin éxito por la existencia de la misma.	El sistema muestra un mensaje de error “El publicación registrada ya existe”. El usuario hace clic izquierdo en el botón Aceptar para cerrar la ventana que contiene el mensaje de información.	SIGAC “Módulo Administración” 1-Hacer clic izquierdo en el menú Indicadores 2-Hacer clic izquierdo en la opción Publicación 3- Hacer clic izquierdo en el botón Registrar Publicación. 4-El sistema muestra la interfaz para Registrar una Publicación 5-El usuario completa los campos especificados 6-Hacer clic izquierdo en el botón Guardar 7-Hacer clic izquierdo en el botón Aceptar.
--	---	---	---

**Tabla 10: Descripción de variable**

Nº	Nombre de campo	Clasificación	Valor nulo	Descripción
1	Autor	Campo de texto	No	Permite insertar el nombre del autor.
3	Título	Campo de texto	No	Permite introducir el usuario del dominio.
4	Categoría	Lista desplegable.	No	Permite seleccionar la categoría de la publicación.
5	Detalles	Campo de texto.	No	Permite introducir detalles de la publicación(indexación o registro)
6	Fecha	Lista desplegable.	No	Permite seleccionar la fecha

Tabla 11: Matriz de datos (SC\_1: Adicionar publicación)

ID del Escenario	Autor	Título	Categoría	Detalles	Fecha	Respuesta del sistema	Resultado de la prueba
EC_1.1: Adicionar publicación con éxito.	V "Dainovy Rodríguez Marrero"	V "Sistema de edición de noticias"	V "memoria"	V " Informática 2013. VI Congreso Internacional de Tecnologías"	V " 12.06-2013"	El sistema permite añadir una nueva publicación	Satisfactorio
EC_1.2: Añadir publicación sin éxito.	V "Liset García Padrón"	V " SyGMe: Sistema para la gestión de metadatos de artículos científicos"	V "serie científica"	V "	V " 27.03-2013"	El sistema señala el campo que no fue completado y muestra el mensaje de error	Satisfactorio
EC_1.3: Cancelar petición.	V "Liset García Padrón"	V " SyGMe: Sistema para la gestión de metadatos de artículos científicos"	V "serie científica"	V "http://publicaciones.uci.cu/index.php/SC/article/view/1083 "	V " 27.03-2013"	El sistema no guarda los cambios.	Satisfactorio
EC_1.4: Añadir publicación sin éxito por la existencia de la misma.	V "Dainovy Rodríguez Marrero"	V "Sistema de edición de noticias"	V "memoria"	V " Informática 2013. VI Congreso Internacional de Tecnologías"	V " 12.06-2013"	El sistema no permite añadir la publicación	Satisfactorio

Después de haber sido aplicados los casos de pruebas al sistema, se detectaron en una primera iteración 10 no conformidades asociadas a errores ortográficos, de interfaz y de validación, las cuales fueron resueltas en período de tiempo de 2 días. Se realizó una segunda iteración y se detectaron 4 no conformidades, asociadas a errores ortográficos y de validación, siendo solucionadas en 1 día. En una tercera iteración no se detectaron no conformidades siendo eliminadas en su totalidad en las dos iteraciones anteriores. En la Figura 6 se muestra una gráfica con la cantidad de no conformidades detectadas en cada iteración.

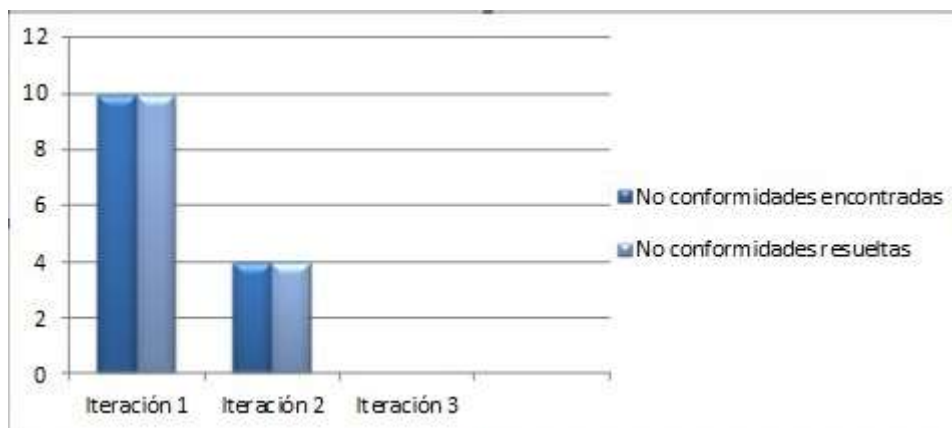


Figura 6: No conformidades por iteración.

### 4.3 Conclusiones parciales

Como resultado del presente capítulo se logró completar la fase de implementación del sistema, a partir de la puesta en práctica de las actividades descritas en las tarjetas CRC. Se le realizaron pruebas al sistema mediante la aplicación de casos de pruebas, los cuales arrojaron resultados positivos permitiendo comprobar que el software satisface las necesidades para las cuales fue desarrollado.

### CONCLUSIONES GENERALES

Una vez finalizada la fundamentación teórica que sustenta la presente investigación, definidas las características del sistema y efectuado su desarrollo y validación, se obtuvieron resultados que permiten arribar a las siguientes conclusiones:

- El estudio del arte de las soluciones analizadas demuestra que estas no permiten darle solución al problema planteado en la investigación, aunque poseen características que sirvieron como base para elaborar la presente solución.
- Se diseñó e implementó una aplicación que gestiona de forma centralizada la información referente a la actividad de investigación y postgrado del centro GEYSED, lo cual permite la organización y control de la misma favoreciendo la toma de decisiones oportuna.
- Una vez realizadas las pruebas de *software*, las cuales arrojaron resultados positivos, se verificó la correcta implementación de las funcionalidades del sistema y se confirma que el mismo cumple con todos los requisitos y que satisface las necesidades del cliente.



### RECOMENDACIONES

Una vez terminado el desarrollo de la aplicación se recomienda:

- Desplegar el sistema en los demás centros de la facultad.
- Incluir una funcionalidad que permita generar de forma automática el Balance de Ciencia y Técnica.

### REFERENCIAS BIBLIOGRÁFICAS

1. **Acuña, Cesar Javier. 2010.** *Arquitectura de software*. 2010.
2. **Arjonilla, Jesus. 2009.** *La gestión de los sistemas de información en la empresa: Teoría y casos prácticos*. España : Ediciones Piramide, 2009.
3. **Acreditación, Junta de Acreditación Nacional. 2013.** *Sistema de acreditación de programas de Maestrías y Doctorado*. 2013.
4. **Bahit, Eugenia. 2011.** *Programación Orientada a Objetos y Modelo-Vista-Controlador en PHP*. s.l. : Autoedición, 2011.
5. **Beck, Kent. 1999.** *“Extreme Programming Explained. Embrace Change”*, Pearson Education, 1999. Traducido al español como: *“Una explicación de la programación extrema. Aceptar el cambio”*, Addison Wesley, 2000.
6. **Calabria, Luis. 2003.** Metodología XP. Cátedra de Ingeniería de Software, Universidad ORT,. [En línea] 2003. [Citado el: 15 de noviembre de 2013.] [http://fi.ort.edu.uy/innovaportal/file/2021/1/metodologia\\_xp.pdf](http://fi.ort.edu.uy/innovaportal/file/2021/1/metodologia_xp.pdf).
7. **CIBERAULA.2010** Una introducción a Apache. 4 de febrero de 2013 2010, nº Disponible en: [http://linux.ciberaula.com/articulo/linux\\_apache\\_intro/](http://linux.ciberaula.com/articulo/linux_apache_intro/).
8. **Eguíluz, Javier. 2008.** *Introducción a Java Script*. 2008.
9. **Eguíluz, Javier. 2009.** *Introducción a CSS*. 2009.
10. **Eguíluz, Javier. 2013.** *Desarrollo web agil con Symfony2*. 2013.
11. **Esteve, J. 2004.** Administración avanzada de GNU/Linux. 3 de febrero de 2013 2004, nº [http://sunshine.prod.uci.cu/gridfs/sunshine/books/Administracin\\_avanzada\\_de\\_GNU\\_Linux.pdf](http://sunshine.prod.uci.cu/gridfs/sunshine/books/Administracin_avanzada_de_GNU_Linux.pdf).
12. **Fabien Potencier, François Zaninotto. 2010.** *Symfony la guía definitiva*. 2010.
13. **Ferrer, Jorge, García, Victor y García, Rodrigo. 1998.** *Curso Completo de HTML*. 1998. Curso.
14. **Figuerola, Roberth y Solís, Camilo J. .2009.** *Metodologías tradicionales vs. Metodologías ágiles*. s.l. : Universidad Técnica Particular de Loja, Escuela de Ciencias en Computación.
15. **González, I. 2007.** Lenguajes del lado servidor y del lado cliente . [En línea] 2007. <http://eats.wordpress.com/2007/01/17/lenguajes-del-lado-servidor-y-del-lado-cliente>.
16. **Greenwald, Rick, Stackowiak, Robert and Stern, Jonathan. 2004.** *Oracle essentials: Oracle database 10g* . s.l. : Third edition Publisher O'Reilly , 2004. Chapter1 Introducing Oracle.

17. **Guid, Administrator's. 2004.** *Administrator's Guide.Chapter 1.General Information.* s.l. : Sams Publishing, 2004.
18. **Inidicadores. 2009.** *Indicadores de medición de la actividad de ciencia y técnica para los centros de educación superior.* [En línea] 19 de enero de 2009. [Citado el: 26 de febrero de 2014.] [https://investigaciones.uci.cu/downloads.php?cat\\_id=17.](https://investigaciones.uci.cu/downloads.php?cat_id=17)
19. **Investigaciones. 2013.** *Política científica de la universidad de las ciencias informáticas.* [En línea] 28 de octubre de 2013. [Citado el: 26 de febrero de 2014.] [https://investigaciones.uci.cu/downloads.php?cat\\_id=18.](https://investigaciones.uci.cu/downloads.php?cat_id=18)
20. **IPLAC. 1990.** Instituto Pedagógico Latinoamericano y Caribeño, Universidad Virtual. [En línea] 1990. [Citado el: 9 de diciembre de 2013.] [http://www.iplac.rimed.cu/.](http://www.iplac.rimed.cu/)
21. **Larman, Pearson. 1999.** *UML y Patrones.* 1999. ISBN 8420534382.
22. **LProgramación. 2012.** Lenguajes de programación . [En línea] 2012. <http://www.frt.utn.edu.ar/sistemas/paradigmas/lenguajes.html>
23. **Medina León, Yordanis. 2012.** *Sistema informático para la gestión de los indicadores de Ciencia e Innovación.* La Habana : UCI, 2012.
24. **Metodología\_FDD. 2012.** [En línea] 12 de junio de 2012. [Citado el: 15 de noviembre de 2013.] [http://metodologiafdd.blogspot.com/2012/06/metodologia-fdd.html.](http://metodologiafdd.blogspot.com/2012/06/metodologia-fdd.html)
25. **Patrones GOF. 2011.** *Patrones del "Gang of Four"* Unidad Docente de Ingeniería del Software. Facultad de informática - Universidad Politécnica de Madrid
26. **PostgreSQL. 2010.** *Global Development Group.PostgreSQL 8.4.4 Documentation.* California : Berkeley, 2010. pp. 368-78, 2013-2033.
27. **Pressman, Roger S. 2005.** *Ingeniería de Software. Un enfoque práctico.* 6ta Edición. s.l. : Editorial McGraw-Hill, 2005. ISBN: 9701054733.
28. **Potencier, 2008.** *Symfony, la guía definitiva.*
29. **Ruiz, Francisco. 2012** *Ingeniería de Software 1.* 5 de febrero de 2012 nº Disponible en: [http://www.ctr.unican.es/asignaturas/is1/is1-t02-trans.pdf.](http://www.ctr.unican.es/asignaturas/is1/is1-t02-trans.pdf)
30. **Sancho, Roberto. 2010.** *Directrices de la Oede para la obtención de Indicadores de Ciencia y Tecnología.* Madrid, España. : Ministerio de Ciencia y Tecnología, 2010.
31. **Scielo. 2010.** *Información Tecnológica.* [En línea] 2010. [Citado el: 4 de abril de 2014.] [http://www.scielo.cl/scielo.php?pid=S0718-07642007000100014&script=sci\\_arttext.](http://www.scielo.cl/scielo.php?pid=S0718-07642007000100014&script=sci_arttext)
32. **Softpedia. 2013.** [En línea] 2013. [Citado el: 10 de mayo de 2014.] <http://www.softpedia.es/programa-Visual-Paradigm-for-UML-Standard-Edition-4139.htm>
33. **Spinak, Ernesto. 1998.** *Indicadores cuantitativos.* 1998.

## ANEXOS

### Anexo I: Entrevista realizada al Ing. Jean Michel Suárez Subdirector de Investigación y Postgrado del centro GEYSED.

1. ¿Cómo se gestiona la actividad científica y de postgrado del centro?
2. ¿Cuáles son los indicadores que componen el Plan de Indicadores y cada que tiempos se confecciona esta planificación?
3. ¿Cuáles son los indicadores cuantitativos que se utilizan para medir la actividad de investigación y postgrado en el centro?
4. ¿Cuáles son los elementos que se tienen en cuenta para medir cada uno de los indicadores?
5. ¿Cuáles son los mecanismos para la recopilación y verificación de la información asociada a la actividad científica y de superación de los profesionales?

### Anexo II: Historias de Usuarios definidas.

Tabla 12: HU\_1 Gestionar Claustro.

Historia de Usuario	
<b>Número:</b> HU_1	<b>Nombre Historia de Usuario:</b> Gestionar Claustro
<b>Modificación de Historia de Usuario Número:</b> Ninguna	
<b>Usuario:</b> Gabriel D. Pérez Fernández	<b>Iteración Asignada:</b> 1
<b>Prioridad en Negocio:</b> Alta	<b>Puntos Estimados:</b> 1 semana
<b>Riesgo en Desarrollo:</b> Alto	<b>Puntos Reales:</b> 1 semana
<b>Descripción:</b> la presente HU tiene como objetivo permitir adicionar, modificar, eliminar y visualizar los datos de un profesional. Se incluirán los datos básicos del profesional como el nombre, apellidos; además de la categoría científica, categoría docente y graduación	
<b>Observaciones:</b> El usuario debe completar los datos que se le solicitan y el sistema valida la correcta entrada de los mismos.	

Tabla 13: HU\_3 Gestionar premios de un profesional.

Historia de Usuario	
<b>Número:</b> HU_3	<b>Nombre Historia de Usuario:</b> Gestionar premios de un profesional
<b>Modificación de Historia de Usuario Número:</b> Ninguna	
<b>Usuario:</b> Gabriel D. Pérez Fernández	<b>Iteración Asignada:</b> 1
<b>Prioridad en Negocio:</b> Alta	<b>Puntos Estimados:</b> 1 semana
<b>Riesgo en Desarrollo:</b> Alto	<b>Puntos Reales:</b> 1 semana
<b>Descripción:</b> la presente HU tiene como objetivo permitir adicionar, modificar, eliminar y visualizar las premios realizadas por un profesional. Se incluirán los datos básicos del profesional como el nombre, apellidos; además de los datos necesarios para el registro de una premio. De un premio se debe conocer: tema, tipo de premio, evento, detalles y fecha	
<b>Observaciones:</b> El usuario debe completar los datos que se le solicitan y el sistema valida la correcta entrada de los mismos.	

Tabla 14: HU\_4 Gestionar capacitación de un profesional.

Historia de Usuario	
<b>Número:</b> HU_4	<b>Nombre Historia de Usuario:</b> Gestionar capacitación de un profesional
<b>Modificación de Historia de Usuario Número:</b> Ninguna	
<b>Usuario:</b> Gabriel D. Pérez Fernández	<b>Iteración Asignada:</b> 1
<b>Prioridad en Negocio:</b> Alta	<b>Puntos Estimados:</b> 1 semana
<b>Riesgo en Desarrollo:</b> Alto	<b>Puntos Reales:</b> 1 semana
<b>Descripción:</b> la presente HU tiene como objetivo permitir adicionar, modificar, eliminar y visualizar la capacitación o recibida por un profesional. Se incluirán los datos básicos del profesional como el nombre, apellidos; además de los datos necesarios para el registro de la capacitación. De la capacitación se debe conocer: tema, tipo, universidad, estado, y fecha.	
<b>Observaciones:</b> El usuario debe completar los datos que se le solicitan y el sistema valida la correcta entrada de los mismos.	

Tabla 15: HU\_5 Gestionar trabajos de un profesional.

Historia de Usuario	
<b>Número:</b> HU_5	<b>Nombre Historia de Usuario:</b> Gestionar trabajos de un profesional.
<b>Modificación de Historia de Usuario Número:</b> Ninguna	
<b>Usuario:</b> Gabriel D. Pérez Fernández	<b>Iteración Asignada:</b> 1
<b>Prioridad en Negocio:</b> Alta	<b>Puntos Estimados:</b> 1 semana
<b>Riesgo en Desarrollo:</b> Alto	<b>Puntos Reales:</b> 1 semana
<b>Descripción:</b> la presente HU tiene como objetivo permitir adicionar, modificar, eliminar y visualizar los trabajos realizadas por un profesional. Se incluirán los datos básicos del profesional como el nombre, apellidos; además de datos necesarios para el registro de un trabajo. De un trabajo se debe conocer: categoría, título, autor, fecha, evento (indexación o ISSN).	
<b>Observaciones:</b> El usuario debe completar los datos que se le solicitan y el sistema valida la correcta entrada de los mismos.	

Tabla 16: HU\_6 Gestionar proyectos I+D del centro.

Historia de Usuario	
<b>Número:</b> HU_6	<b>Nombre Historia de Usuario:</b> Gestionar proyectos I+D del centro.
<b>Modificación de Historia de Usuario Número:</b> Ninguna	
<b>Usuario:</b> Gabriel D. Pérez Fernández	<b>Iteración Asignada:</b> 1
<b>Prioridad en Negocio:</b> Alta	<b>Puntos Estimados:</b> 1 semana
<b>Riesgo en Desarrollo:</b> Alto	<b>Puntos Reales:</b> 1 semana
<b>Descripción:</b> la presente HU tiene como objetivo permitir adicionar, modificar, eliminar y visualizar los proyectos I+D (Investigación y Desarrollo) del centro. De un proyectos I+D se gestionan los siguientes datos: categoría (facultad, territorial, universitarios, nacional, internacional), nombre del proyecto, fase en la que se encuentra, estado de cumplimiento (detenido o en ejecución), resultados obtenidos y si posee aporte tecnológico.	
<b>Observaciones:</b> El usuario debe completar los datos que se le solicitan y el sistema valida la correcta entrada de los mismos.	

Tabla 17: HU\_7 Gestionar patentes y registros del centro.

Historia de Usuario	
<b>Número:</b> HU_7	<b>Nombre Historia de Usuario:</b> Gestionar patentes y registros del centro.
<b>Modificación de Historia de Usuario Número:</b> Ninguna	
<b>Usuario:</b> Gabriel D. Pérez Fernández	<b>Iteración Asignada:</b> 1
<b>Prioridad en Negocio:</b> Alta	<b>Puntos Estimados:</b> 1 semana
<b>Riesgo en Desarrollo:</b> Alto	<b>Puntos Reales:</b> 1 semana
<b>Descripción:</b> la presente HU tiene como objetivo permitir adicionar, modificar, eliminar y visualizar los patentes y registros del centro. De las patentes y registros se gestionan los siguientes datos: producto tipo, entidad y fecha.	
<b>Observaciones:</b> El usuario debe completar los datos que se le solicitan y el sistema valida la correcta entrada de los mismos.	

Tabla 18: HU\_8 Gestionar resultados introducidos.

Historia de Usuario	
<b>Número:</b> HU_8	<b>Nombre Historia de Usuario:</b> Gestionar resultados introducidos.
<b>Modificación de Historia de Usuario Número:</b> Ninguna	
<b>Usuario:</b> Gabriel D. Pérez Fernández	<b>Iteración Asignada:</b> 1
<b>Prioridad en Negocio:</b> Alta	<b>Puntos Estimados:</b> 1 semana
<b>Riesgo en Desarrollo:</b> Alto	<b>Puntos Reales:</b> 1 semana
<b>Descripción:</b> la presente HU tiene como objetivo permitir adicionar, modificar, eliminar y visualizar los resultados introducidos. De resultados introducidos se gestionan los siguientes datos: producto tipo, departamento y fecha.	
<b>Observaciones:</b> El usuario debe completar los datos que se le solicitan y el sistema valida la correcta entrada de los mismos.	

Tabla 19: HU\_9 Gestionar categorías.

Historia de Usuario	
Número: HU_9	Nombre Historia de Usuario: Gestionar categorías.
Modificación de Historia de Usuario Número: Ninguna	
Usuario: Gabriel D. Pérez Fernández	Iteración Asignada: 1
Prioridad en Negocio: Alta	Puntos Estimados: 1 semana
Riesgo en Desarrollo: Alto	Puntos Reales: 1 semana
<b>Descripción:</b> la presente HU tiene como objetivo permitir adicionar, modificar, eliminar y visualizar las categorías que pueden tener los indicadores.	
<b>Observaciones:</b> El usuario debe completar los datos que se le solicitan y el sistema valida la correcta entrada de los mismos.	

Tabla 20: HU\_10 Autenticar Usuario.

Historia de Usuario	
Número: HU_10	Nombre Historia de Usuario: Autenticar Usuario
Modificación de Historia de Usuario Número: Ninguna	
Usuario: Gabriel D. Pérez Fernández	Iteración Asignada: 1
Prioridad en Negocio: Alta	Puntos Estimados: 1 semana
Riesgo en Desarrollo: Alto	Puntos Reales: 1 semana
<b>Descripción:</b> la presente HU tiene como objetivo permitir la autenticación de los usuarios para acceder el sistema.	
<b>Observaciones:</b> El usuario debe completar los datos que se le solicitan y el sistema valida la correcta entrada de los mismos.	



Tabla 21: HU\_11 Gestionar departamento.

Historia de Usuario	
<b>Número:</b> HU_11	<b>Nombre Historia de Usuario:</b> Gestionar departamento.
<b>Modificación de Historia de Usuario Número:</b> Ninguna	
<b>Usuario:</b> Gabriel D. Pérez Fernández	<b>Iteración Asignada:</b> 1
<b>Prioridad en Negocio:</b> Alta	<b>Puntos Estimados:</b> 1 semana
<b>Riesgo en Desarrollo:</b> Alto	<b>Puntos Reales:</b> 1 semana
<b>Descripción:</b> la presente HU tiene como objetivo permitir adicionar, modificar, eliminar y visualizar los datos de los departamentos.	
<b>Observaciones:</b> El usuario debe completar los datos que se le solicitan y el sistema valida la correcta entrada de los mismos.	
<b>Prototipo de interface:</b>	

Tabla 22: HU\_12 Gestionar centro.

Historia de Usuario	
<b>Número:</b> HU_12	<b>Nombre Historia de Usuario:</b> Gestionar centro
<b>Modificación de Historia de Usuario Número:</b> Ninguna	
<b>Usuario:</b> Gabriel D. Pérez Fernández	<b>Iteración Asignada:</b> 1
<b>Prioridad en Negocio:</b> Alta	<b>Puntos Estimados:</b> 1 semana
<b>Riesgo en Desarrollo:</b> Alto	<b>Puntos Reales:</b> 1 semana
<b>Descripción:</b> la presente HU tiene como objetivo permitir adicionar, modificar, eliminar y visualizar los datos de los centros.	
<b>Observaciones:</b> El usuario debe completar los datos que se le solicitan y el sistema valida la correcta entrada de los mismos.	
<b>Prototipo de interface:</b>	

Anexo III: Tarjetas CRC confeccionadas.

Tabla 23: Tarjeta CRC Clase Publicación

Tarjeta CRC	
Publicacion.php	
Responsabilidades	Colaboradores
Se implementan los métodos para la creación, modificación, visualización y eliminación de las publicaciones realizadas por un profesional.	PublicacionController.php
Contiene el constructor del formulario que permite la inserción de los datos para la posterior creación, modificación y visualización de la información asociada a publicaciones realizadas por un profesional.	PublicacionType.php

Tabla 24: Tarjeta CRC Clase Trabajo

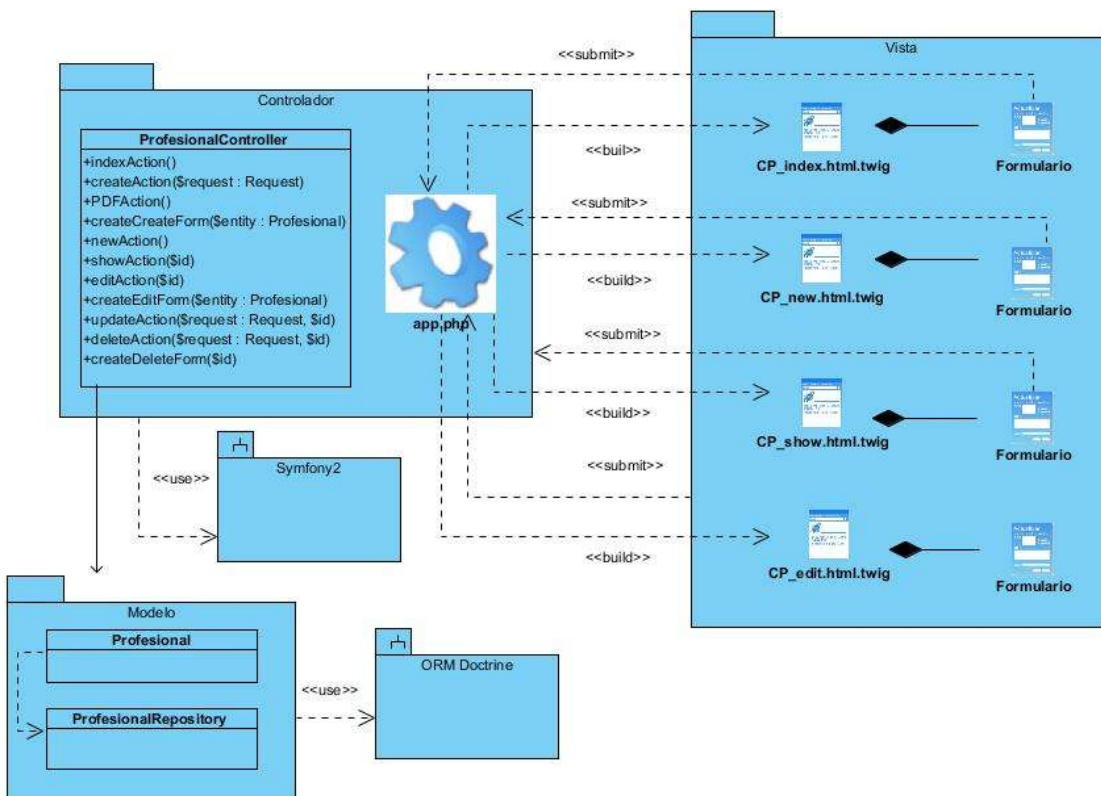
Tarjeta CRC	
Trabajo.php	
Responsabilidades	Colaboradores
Se implementan los métodos para la creación, modificación, visualización y eliminación de los trabajos realizadas por un profesional.	TrabajoController.php
Contiene el constructor del formulario que permite la inserción de los datos para la posterior creación, modificación y visualización de la información asociada a los trabajos realizados por un profesional.	TrabajoType.php

Tabla 25: Tarjeta CRC Clase ProyectoID

Tarjeta CRC	
ProyectoID.php	
Responsabilidades	Colaboradores
Se implementan los métodos para la creación, modificación, visualización y eliminación de los	ProyectoIDController.php

<p>proyectos I+D ejecutados por el centro.</p>	
<p>Contiene el constructor del formulario que permite la inserción de los datos para la posterior creación, modificación y visualización de la información asociada a los proyectos I+D ejecutados por el centro.</p>	<p>ProyectoIDType.php</p>

**Anexo IV: Diagramas de clases del diseño.**



**Figura 7: Diagrama de clases del diseño, HU\_1: Gestionar claustro.**

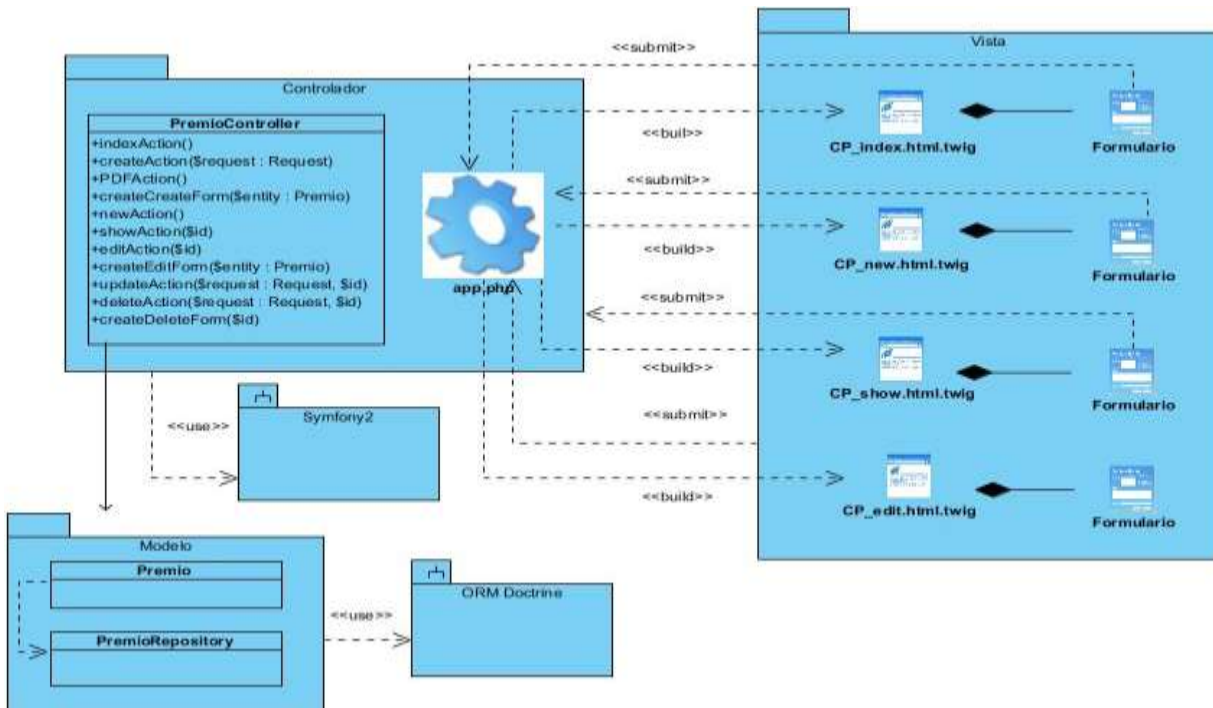


Figura 8: Diagrama de clases del diseño, HU\_3: Gestionar premios de un profesional.

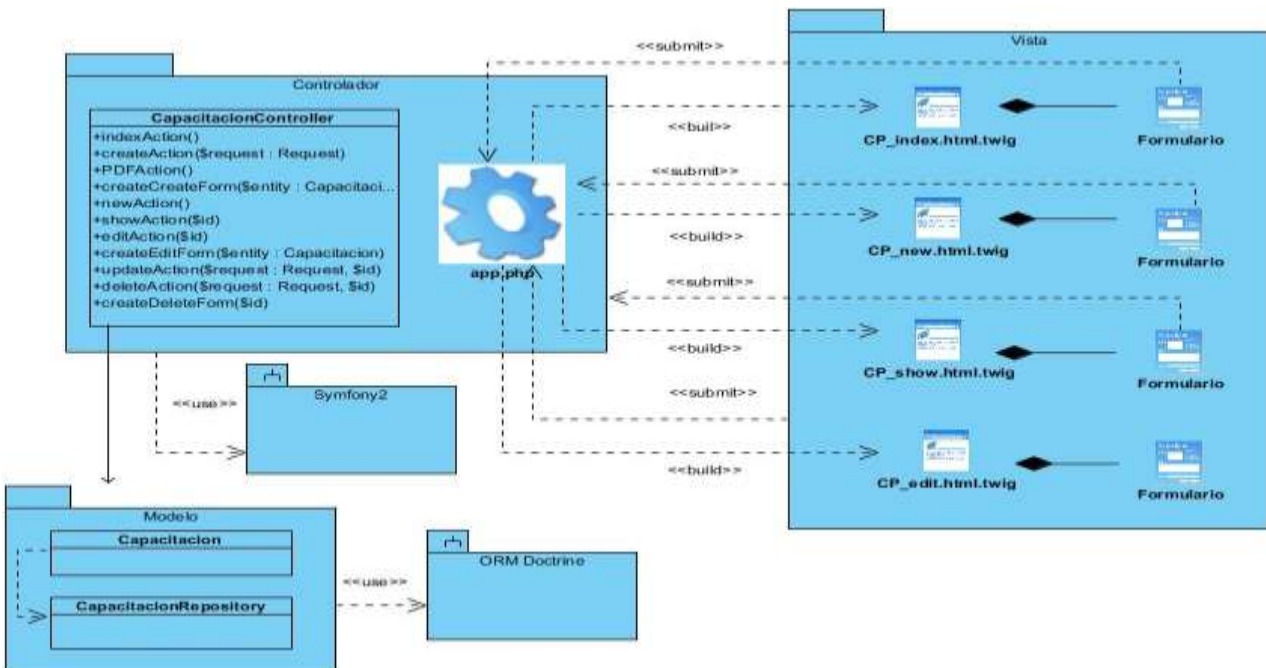


Figura 9: Diagrama de clases del diseño, HU\_3: Gestionar capacitación de un profesional.

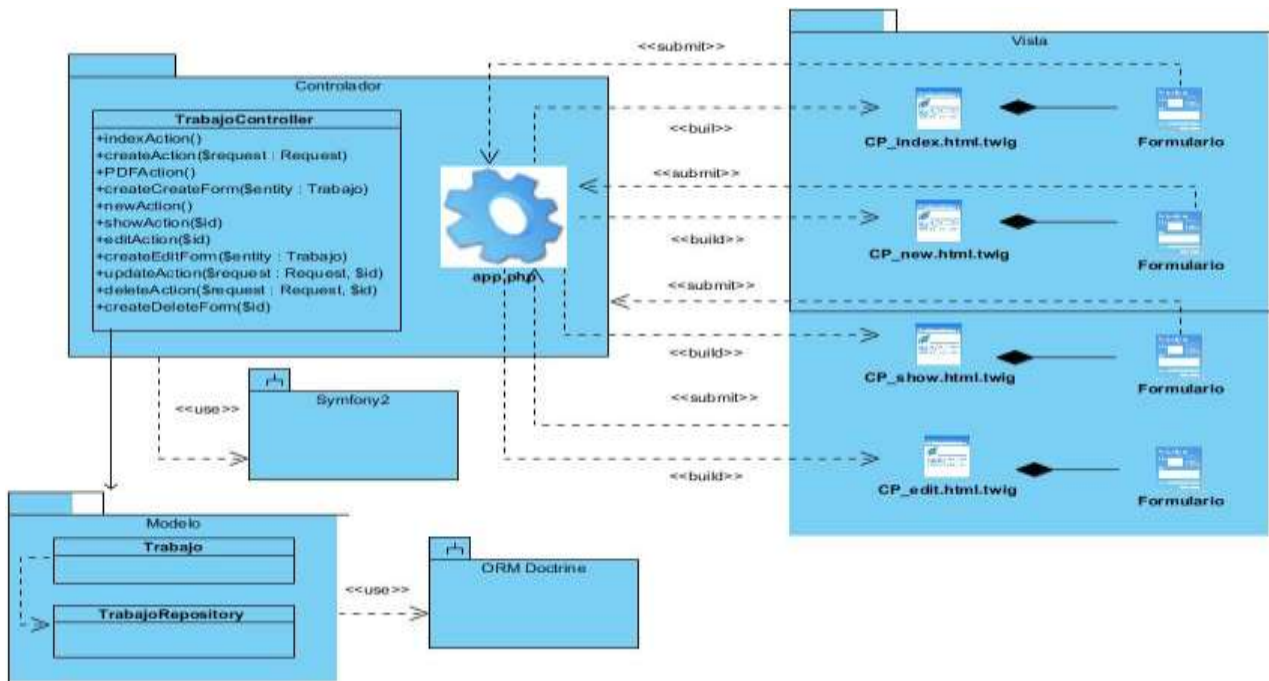


Figura 10: Diagrama de clases del diseño, HU\_5: Gestionar trabajos de un profesional.

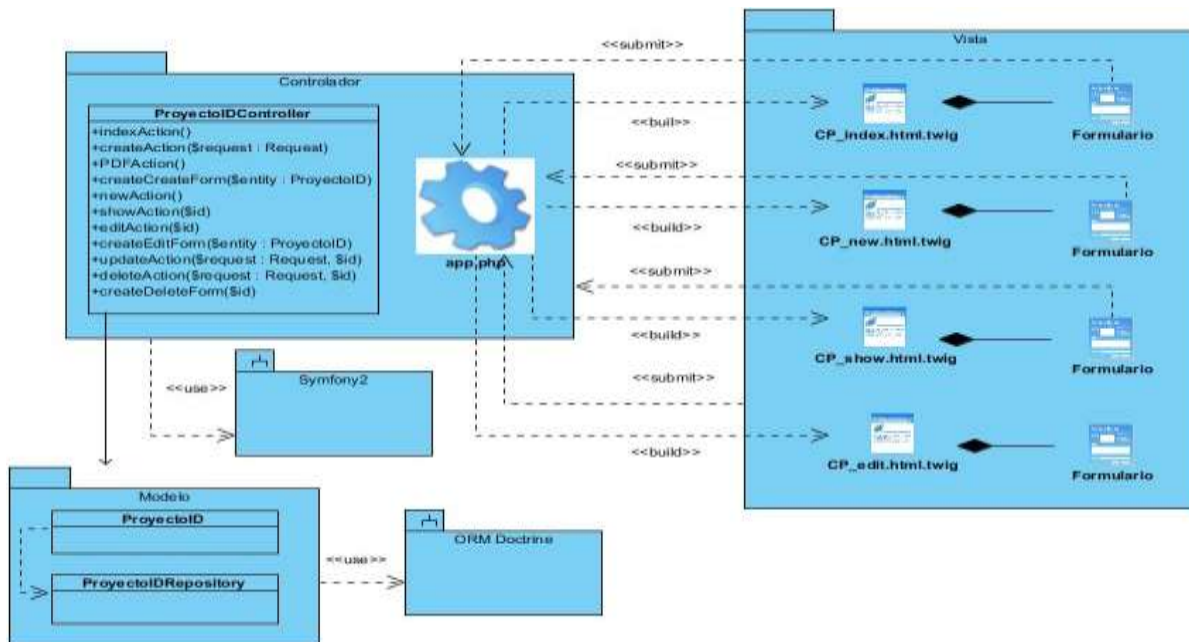


Figura 11: Diagrama de clases del diseño, HU\_6: Gestionar proyecto I+D del centro.



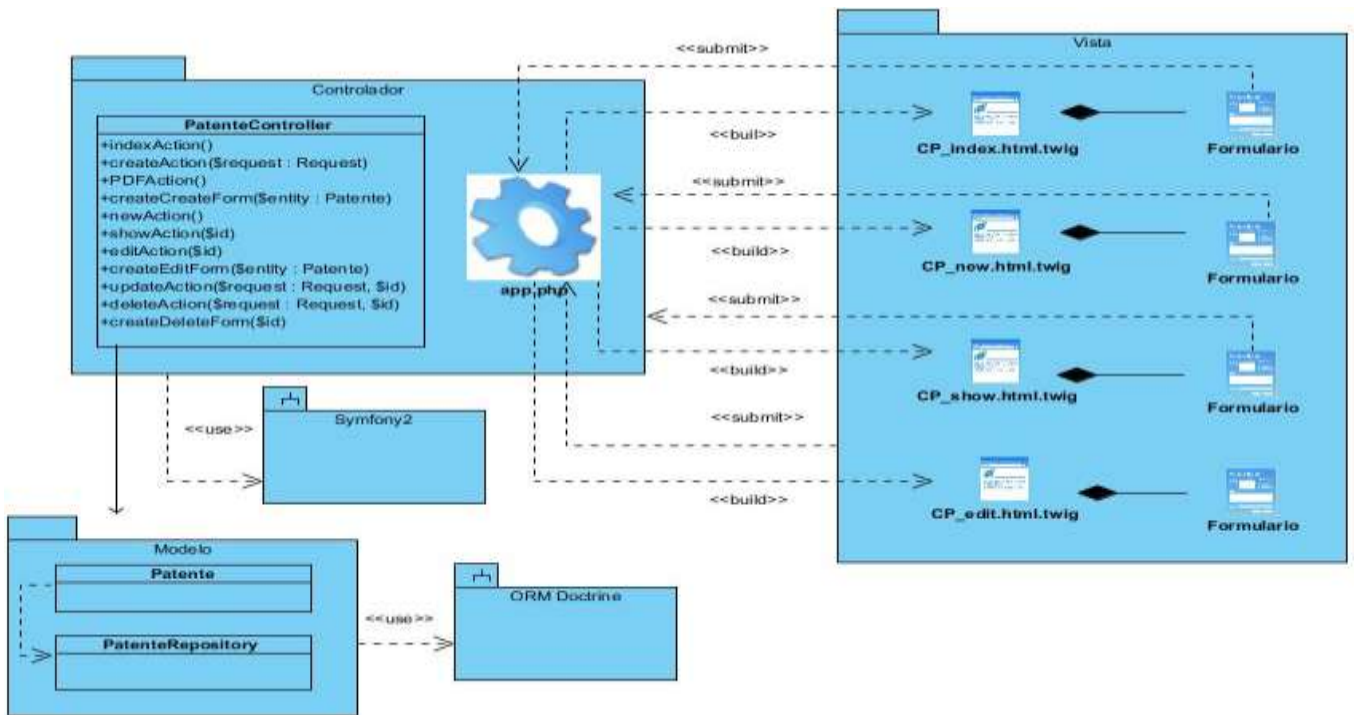


Figura 12: Diagrama de clases del diseño, HU\_7: Gestionar patentes y registros del centro.

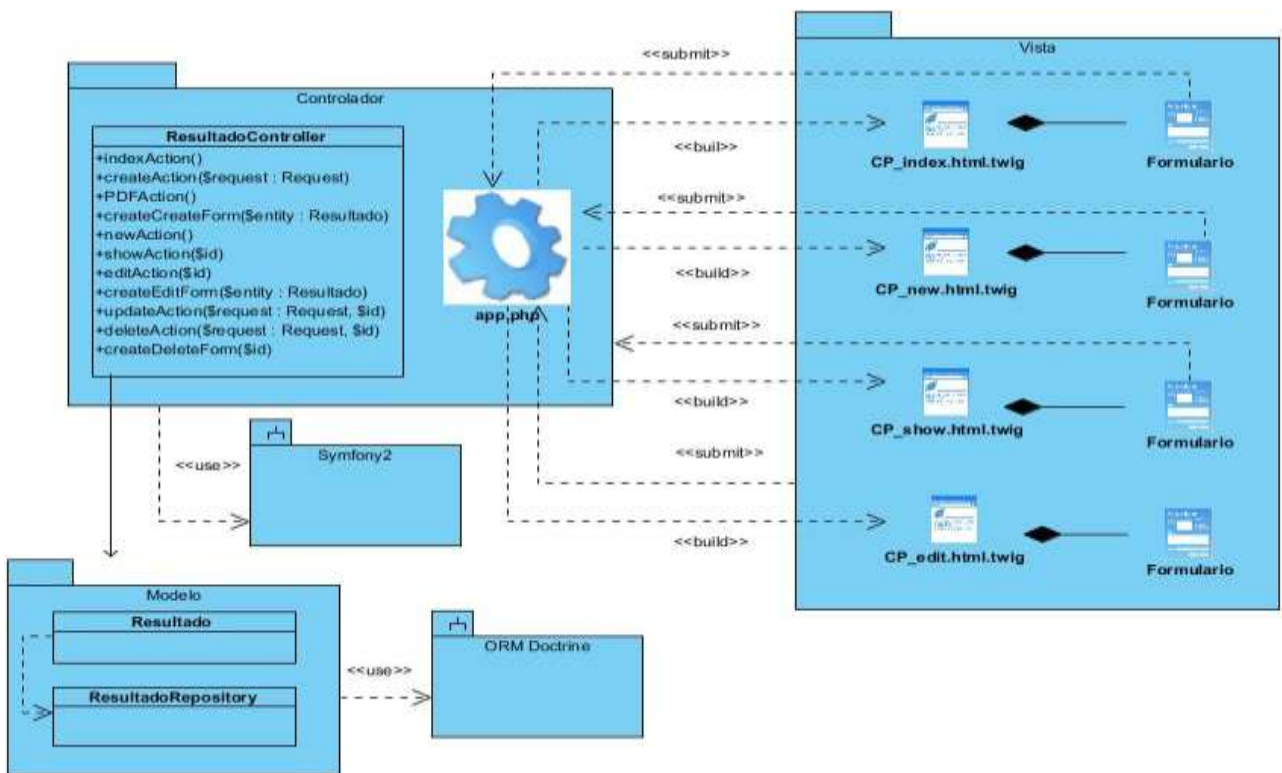


Figura 13: Diagrama de clases del diseño, HU\_8: Gestionar resultados introducidos en el centro.