

Universidad de las Ciencias Informáticas  
FACULTAD 1

---

# Sistema para la gestión del banco de problemas científicos de la Facultad 1

---

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

**Autora:**

Ana Carla Del Real Baró

**Tutores:**

Msc. Ailyn Gutiérrez Ferrera

Ing. Adyana Pileta Gammalane

**La Habana, 2019**

**“Año 61 de la Revolución”**





**“SOMOS ARQUITECTOS DE  
NUESTRO PROPIO DESTINO”**

*-Albert Einstein*

## DECLARACIÓN DE AUTORÍA

Declaro por este medio que yo Ana Carla Del Real Baró, con carnet de identidad 96110600613, soy el autor principal del trabajo titulado “**Sistema para la gestión del banco de problemas científicos de la Facultad 1**” y autorizo a la Universidad de las Ciencias Informáticas a hacer uso del mismo en su beneficio, así como los derechos patrimoniales con carácter exclusivo.

Declaro que todo lo anteriormente expuesto se ajusta a la verdad, y asumo la responsabilidad moral y jurídica que se derive de este juramento profesional.

Y para que así conste, firmo la presente declaración de autoría en La Habana a los \_\_\_\_\_ días del mes de \_\_\_\_\_ del año 2019.

**Autora:**

\_\_\_\_\_

Ana Carla Del Real Baró

**Tutores:**

\_\_\_\_\_

Msc. Ailyn Gutiérrez Ferrera

\_\_\_\_\_

Ing. Adyana Pileta Gammalame

## Resumen

La presente investigación tiene como objetivo desarrollar un sistema que permita realizar una adecuada gestión de las distintas problemáticas de carácter científico originadas en la Facultad 1 de la Universidad de Ciencias Informáticas (UCI). El estudio y análisis del estado del arte permitieron identificar funcionalidades y tecnologías que facilitaron la comprensión y diseño de la solución propuesta. El desarrollo estuvo guiado por la metodología AUP-UCI y se seleccionó como principales tecnologías el marco de trabajo *Symfony*, *PHP* como lenguaje de programación, el entorno integrado de desarrollo *Netbeans* y como gestor de base de datos se utilizó *MySQL*. La estrategia de prueba aplicada permitió verificar el cumplimiento de los objetivos trazados y evaluar la calidad del sistema. El método *Delphi* validó que la solución desarrollada tiene un alto valor para los encargados de asignar los temas de tesis ya que facilita este proceso y mejora la gestión de los problemas de investigación.

**Palabras clave:** gestión, problemas de investigación, temas de tesis, sistema de gestión.

# ÍNDICE

INTRODUCCIÓN .....	3
Capítulo 1: Gestión de la información .....	9
1.1 Conceptos básicos relacionados con el dominio de la investigación .....	9
1.2 Principales sistemas homólogos.....	10
1.2.1 Sistemas de gestión de la información en el ámbito Internacional.....	10
1.2.2 Sistemas de gestión de la información en el ámbito nacional.....	13
1.2.3 Resultados del estudio de los sistemas homólogos .....	14
1.3 Descripción de los lenguajes, herramientas y tecnologías utilizadas .....	14
1.3.1 Marco de trabajo .....	14
1.3.2 Servidor de Base de Datos .....	16
1.3.3 Servidor web.....	16
1.3.4 Lenguaje y Herramienta de Modelado.....	17
1.3.5 Lenguajes de programación.....	17
1.3.6 Lenguaje de programación del lado del servidor.....	18
1.3.7 Lenguaje de programación del lado del cliente .....	18
1.3.8 Entorno Integrado de Desarrollo .....	19
1.4 Metodología de desarrollo .....	19
1.5 Herramientas de validación .....	20
1.6 Conclusiones del capítulo.....	21
Capítulo 2: Análisis y diseño del sistema para la gestión del banco de problemas científicos de la Facultad	
1. ....	22
2.1 Modelo Conceptual.....	22
2.2 Descripción de la propuesta de solución .....	23
2.3 Requisitos de la propuesta de solución .....	24
2.3.1 Requisitos Funcionales .....	24
2.3.2 Requisitos No Funcionales.....	25
2.4 Arquitectura de Software .....	26
2.5 Patrones de Diseño .....	27

2.5.1 Patrones Generales de Software para la Asignación de Responsabilidades (GRASP) .....	28
2.5.2 Patrones Gang of Four (GOF).....	28
2.6 Historias de usuario.....	29
2.7 Diagrama de clases de diseño .....	31
2.8 Diagrama de secuencia.....	33
2.9 Modelo de datos.....	35
2.10 Modelo de despliegue .....	36
2.11 Conclusiones del capítulo.....	37
Capítulo 3: Implementación y validación del sistema para la gestión del banco de problemas científicos de la Facultad 1. ....	38
3.1 Diagrama de Componentes.....	38
3.2 Estándares de Codificación .....	40
3.3 Validación de la propuesta de solución.....	41
3.3.1 Pruebas Funcionales .....	42
3.3.2 Pruebas de Seguridad .....	44
3.3.3 Pruebas de Integración .....	45
3.3.4 Pruebas de Rendimiento.....	46
3.3.5 Validación de la hipótesis científica de la investigación.....	48
3.4 Conclusiones del capítulo.....	53
Conclusiones .....	55
Recomendaciones .....	56
Bibliografía.....	57
Anexos.....	62
Anexo 1. Historias de Usuario .....	62
Anexo 2. Diagramas de clase de diseño.....	69
Anexo 3. Diagramas de secuencia .....	70
Anexo 4. Casos de Prueba.....	71
Anexo 5. Encuesta para la identificación de posibles expertos.....	73
Anexo 6. Encuesta a expertos.....	75

## INTRODUCCIÓN

La información es la base del conocimiento y la vía fundamental que tienen las personas, las organizaciones y los países para comunicarse. Gestionar la información en la actualidad ha tomado prioridad, pues es mayor la cantidad de información que se necesita manejar. Durante los primeros años del siglo XXI esta necesidad ha tenido un impacto no solo en los métodos y técnicas de gestión de la información, también en la propia tecnología para la gestión de la información (Castellanos, 2001).

El conocimiento es un elemento significativo que las organizaciones tienen que ser capaces de administrar y preservar para utilizar en su beneficio, con acciones específicas de intercambio y colaboración. La puesta en práctica de la gestión del conocimiento permite concebir proyectos eficientes, efectivos y eficaces, que contribuyen a precisar resultados con calidad, para el uso eficiente de los recursos y medios que poseen (Sánchez, 2016).

Las aportaciones de la gestión del conocimiento dependen de las líneas estratégicas de la organización. La adecuada explotación de las herramientas de software y la gestión de conocimiento, puede lograr metodologías de enseñanza-aprendizaje en entornos virtuales desde la óptica de elaboración colaborativa y el intercambio de conocimiento (Carbonell, 2016).

La gestión del conocimiento, permite realizar eficazmente el trabajo encomendado a la organización. Mediante su correcta gestión, las organizaciones favorecen que el individuo se desarrolle en su trabajo y aporte ideas, al mismo tiempo evita la "fuga de conocimiento" cuando las personas abandonan la organización. La gestión de la información es imprescindible, pero solo se convierte en conocimiento cuando los individuos la aplican para la resolución de un problema (Iglesias y Ruesta 2001).

En una sociedad basada en el conocimiento, la universidad se convierte en un elemento clave. Su desafío se centra en cómo identificar, medir y evaluar el capital intelectual, para orientar la gestión del conocimiento con el fin de crear nuevas riquezas o mejorar el valor intelectual, lo que contribuirá a diseñar y desplegar políticas científicas y planes de investigación con mayores dosis de eficiencia. Según Iglesias y Ruesta (2001): las universidades ocupan un lugar prominente entre los principales generadores de conocimiento en la sociedad. El desarrollo e incorporación constante del nuevo conocimiento a la vida



diaria es un factor crítico para asegurar el desarrollo de sociedades con la capacidad de responder a los cambios del entorno actual y de construir su futuro.

La integración de la gestión del conocimiento, la ciencia, la tecnología y la innovación como brújula de la gestión educativa estratégica en el ámbito universitario ha acabado con los antiguos límites que abrían brechas entre la investigación, el postgrado y la extensión universitaria. Estas fronteras se vuelven obsoletas o devienen irrelevantes en el marco de la aplicación de modelos interactivos de investigación y postgrado en las universidades.

Un sistema de gestión ayuda a lograr los objetivos de una organización mediante una serie de estrategias, que incluyen la optimización de procesos, el enfoque centrado en la gestión y el pensamiento disciplinado. Existen en el mundo diversos sistemas de gestión: desde un simple registro manual de la correspondencia, hasta sofisticados sistemas informáticos que manejan no solo la documentación administrativa (papel o formato electrónico), además controlan los flujos de trabajo del proceso de tramitación de expedientes, capturan información desde bases de datos, bibliotecas, centros de documentación y permiten realizar búsquedas sofisticadas y recuperar información (Naranjo, 2015).

Se ha venido experimentando un incremento en la utilización de las tecnologías de la información en los últimos años como parte del programa para informatizar la sociedad cubana. La Industria de Software en Cuba a través de sistemas informáticos, utilizando la gestión de la información, ha ido en aumento con el fin de alcanzar una mejora sustancial en la infraestructura tecnológica. Permitiendo de esta forma satisfacer las necesidades de acceso, almacenamiento y utilización de la información, elevando la calidad de vida, la eficiencia y eficacia de los servicios prestados. Este proceso se realiza de manera acelerada y alcanza resultados satisfactorios en áreas esenciales como la Educación, la Salud y la Investigación.

La Universidad de Ciencias Informáticas (UCI), en colaboración con el proceso de informatización de los servicios, ha creado varios sistemas de gestión como solución a diferentes problemáticas. Específicamente en el ámbito del quehacer científico aún queda mucho por hacer con el objetivo de mejorar y agilizar el proceso de gestión para la formación de postgrado. En la facultad 1 de la Universidad de las Ciencias Informáticas en ocasiones resulta engorroso para el Vicedecano de Investigación y Postgrado llevar un control y seguimiento adecuado de los temas de investigación que

desarrollan los trabajadores debido a que la información al respecto se encuentra dispersa y cambia continuamente.

Normalmente el proceso de actualización de la información científica relacionada con el postgrado se realiza solo a inicios de cada año. Dicho proceso de actualización se realiza a través de la solicitud a los directivos administrativos de las diferentes áreas de la facultad y de la información pertinente. Como parte de la información recogida se incluye el tema de investigación de los estudiantes de maestría y doctorado, pero no se incluye el problema de investigación que desarrolla cada investigador en su área. Todo este proceso lleva tiempo y depende de la premura que cada área le imponga. Una vez recogida la información el Vicedecano de Investigación y Postgrado deberá agrupar todos los datos manualmente para su seguimiento y control. Esta información forma parte de las variables que se tienen en cuenta para elaborar los objetivos de trabajo de la facultad para el año en el área del postgrado pero no se actualiza hasta el siguiente año.

Por otro lado, en una universidad no solo se realiza investigación científica desde el postgrado, el pregrado también tiene un protagonismo importante, siendo el proceso de desarrollo de la tesis de Diploma el principal exponente. Cada año la dirección de la facultad se enfrenta a la difícil e importante tarea de asignar temas de investigación a los estudiantes para que lleven a cabo el ejercicio de culminación de estudios. En ocasiones se asignan temas que no responden a las necesidades de la facultad y no hay una adecuada concordancia con las líneas de investigación de la universidad. Al no contar con un banco de problemas de investigación actualizado la asignación de los mismos a estudiantes se ve afectada y como el proceso se hace de manera manual actualmente trae consigo que pudieran cometerse errores de duplicidad y lentitud en el proceso de asignación. Además, en la Facultad 1 no se ha logrado corresponder los problemas de investigación conducentes a tesis de Doctorado y Maestrías con las tesis de pregrado, lo que afecta el establecimiento de una adecuada pirámide de investigación.

Por lo anteriormente expuesto se plantea como **problema de la investigación**: ¿Cómo mejorar el proceso de gestión de la información científica asociada a los problemas de investigación que se desarrollan en la Facultad 1 de la Universidad de las Ciencias Informáticas?

Se define como **objeto de estudio**: El proceso de gestión de problemas de investigación y el **campo de acción** se enmarca en el proceso de gestión de problemas de investigación para el banco de problemas de investigación de la Facultad 1.

Se propone como **objetivo general**: Desarrollar un sistema informático para la gestión de problemas de investigación que permita mejorar el proceso de gestión de la información científica asociada a los problemas de investigación que se desarrollan en la Facultad 1 de la Universidad de las Ciencias Informáticas.

**Objetivos específicos:**

1. Construir los referentes teóricos fundamentales que sustentan la investigación relacionados con la creación de un sistema para la gestión del banco de problemas de investigación de la Facultad 1.
2. Identificar las funcionalidades del Sistema para la gestión del banco de problemas de investigación de la Facultad 1.
3. Implementar las funcionalidades del Sistema para la gestión del banco de problemas de investigación de la Facultad 1.
4. Validar las funcionalidades del Sistema para la gestión del banco de problemas de investigación de la Facultad 1.

Se plantea como **hipótesis científica** que: El desarrollo de un sistema para la gestión de los problemas de investigación mejorará el proceso de gestión de la información científica asociada a los problemas de investigación que se desarrollan en la Facultad 1 de la Universidad de las Ciencias Informáticas.

**Variable independiente:** Sistema para la gestión del banco de problemas de investigación de la Facultad 1.

**Variable dependiente:** Mejorar el proceso de gestión de los problemas de investigación, refiriéndose a la disminución del tiempo, esfuerzo y facilidad en las tareas de gestión del sistema.

**Tareas de investigación:**

1. Realización de estudio del estado del arte sobre la gestión de los problemas de investigación del centro.

2. Caracterización de los sistemas que permitan gestionar problemas de investigación.
3. Levantamiento de información.
4. Selección de las tecnologías, herramientas y estándares que se necesitan para implementar la propuesta de solución.
5. Definición de la arquitectura de software, selección de patrones de diseño y arquitectura.
6. Implementación del Sistema para la gestión del banco de problemas de investigación de la Facultad 1.
7. Validación del Sistema para la gestión del banco de problemas de investigación de la Facultad 1.

Con el propósito de realizar las tareas planteadas para el desarrollo de la investigación se utilizaron los **métodos de investigación** siguientes:

#### **Métodos Teóricos:**

**Histórico-Lógico:** Para analizar el desarrollo del proceso de gestión del banco de problemas de investigación a través del tiempo de proyectos informáticos de este tipo (Sistema para la Gestión del banco de Problemas de Investigación).

**Analítico-Sintético:** Empleado en el estudio de las bibliografías y conceptos existentes, permitiendo extraer los elementos más importantes vinculados al proceso de gestión de problemas de investigación en general.

#### **Métodos Empíricos:**

**Entrevista:** Se realiza con el objetivo de adquirir información sobre la gestión de los problemas de investigación de la Facultad 1, para la determinación de las posibles funcionalidades y otras características que debía tener la aplicación a desarrollar.

#### **Estructura Del Documento**

El presente trabajo de diploma está estructurado de la manera siguiente: introducción, tres capítulos, conclusiones generales, recomendaciones, bibliografía, anexos y glosario de términos.

### **Capítulo 1 Fundamentación Teórica**

En este capítulo se puntualizan los principales conceptos asociados al dominio de la investigación y se realiza un estudio del estado del arte de las soluciones existentes a nivel nacional e internacional. Se estudian, además, las herramientas, metodologías y técnicas utilizadas para dar solución al problema planteado.

### **Capítulo 2 Análisis y Diseño**

En este capítulo se describe la propuesta de solución del sistema a desarrollar y los procesos que serán informatizados con el desarrollo de esta aplicación. También se presentan los requisitos funcionales y no funcionales con los que debe cumplir el sistema propuesto. Se define la arquitectura del sistema, patrones de diseño y los artefactos generados según la metodología utilizada.

### **Capítulo 3 Implementación y Prueba**

En este capítulo se detalla la propuesta de solución al problema planteado. Se describe la organización del sistema en un diagrama de componentes y se especifican los estándares de codificación a utilizar. Se realizan las estrategias de pruebas definidas para el sistema y se muestran interfaces como parte del resultado final.

## Capítulo 1: Gestión de la información

Para comprender la investigación, es necesario describir los aspectos teóricos relacionados con el conocimiento científico, la gestión de la información y las bases técnicas a tener en consideración para el desarrollo del sistema. En el presente capítulo se exponen los fundamentos teóricos asociados al objeto de estudio y campo de acción y se describen brevemente los sistemas utilizados con fines similares. Se caracteriza la metodología de desarrollo escogida y se fundamenta el ambiente de desarrollo utilizado para dar respuesta a la propuesta de solución.

### 1.1 Conceptos básicos relacionados con el dominio de la investigación

Un **Sistema de Gestión (SG)** es una serie de **procesos, acciones y tareas** que se realizan sobre un conjunto de elementos (personas, procedimientos, estrategias, planes, recursos y productos) **para lograr el éxito sostenido de una organización**, disponer de capacidad para satisfacer las necesidades y las expectativas de sus clientes o beneficiarios, trabajadores y de otras partes interesadas a largo plazo y de un modo equilibrado y sostenible (Naranjo, 2015).

**Gestión de la información (GI)** es la denominación convencional de un conjunto de procesos para controlar el ciclo de vida de la información, desde su obtención (por creación o captura), hasta su disposición final (su archivo o eliminación). Estos procesos también comprenden la extracción, combinación, depuración y distribución de la información a los interesados. El objetivo de la gestión de la información es garantizar la integridad, disponibilidad y confidencialidad de la información. En el contexto de las organizaciones, la gestión de la información se puede identificar como la disciplina que se encarga de todo lo relacionado con la obtención de la información adecuada, en la forma correcta, para la persona indicada, al coste adecuado, en el momento oportuno, en el lugar apropiado y articulando todas estas operaciones para el desarrollo de una acción correcta. En este contexto, los objetivos principales de la GI son: maximizar el valor y los beneficios derivados del uso de la información, minimizar el coste de adquisición, procesamiento y uso de la información, determinar responsabilidades para el uso efectivo, eficiente y económico de la información y asegurar un suministro continuo de la información (Horruitiner, 2017).

Los **Bancos de Información** generalmente son bases de datos. Almacenan datos proporcionados por un usuario, permiten acceder a la información de manera rápida y segura (Espinoza, 2012)

La **Gestión del Conocimiento (GC)** significa organizar la información y el conocimiento de una organización de manera holística. Al principio en el movimiento de GC, (Davenport,1998) ofreció la definición ampliamente citada:

“La gestión del conocimiento es el proceso de captura, distribución y uso eficaz del conocimiento”. (Ramírez, 2017)

Es importante destacar que la noción de **problemas de investigación** no suele referirse a un inconveniente que surge en la labor investigativa, sino que está vinculada al fenómeno específico que se pretende investigar. Un problema de investigación se explica a partir de la tarea del investigador. Su acepción como “*problema*” se vincula a que la finalidad de una investigación es brindar una solución: lo que se investiga supone un problema a resolver. Se trata, del **porqué de la investigación**, que aparece tras el diagnóstico que el investigador realiza acerca de un fenómeno, un proceso o un hecho que no puede explicar o que desconoce sus causas y/o efectos (Julián Pérez Porto, 2015).

## **1.2 Principales sistemas homólogos**

Debido a la importancia que se le confiere a la información, hoy existen varias herramientas creadas con el objetivo de apoyar el proceso de gestión de la misma. Es por ello que se decide hacer un análisis de varios sistemas desarrollados a nivel nacional e internacional con el objetivo de determinar las principales características y funcionalidades presentes en estos sistemas similares. Así como sus ventajas y desventajas que puedan favorecer el desarrollo de la investigación, a continuación se muestran algunas de ellas:

### **1.2.1 Sistemas de gestión de la información en el ámbito Internacional**

#### **HelpDesk**

Es un sitio web donde se gestionan incidencias, tickets y solicitudes de distintas áreas de trabajo. Toda la información quedará almacenada en el HelpDesk, lo que hará que su equipo trabaje más rápido. Tiene como limitante que el sistema es online, lo que crea una dependencia del internet. Ofrece varios tipos de informes diferentes para el análisis y seguimiento de su servicio de soporte, define avisos por correo electrónico y notificaciones para un mejor seguimiento y cumplimiento del Acuerdo de Nivel de Servicio (ANS). Administra las solicitudes recibidas y deja en manos del usuario la decisión de qué miembro del equipo de soporte gestionará cada incidencia. Posee una visión general de todas las incidencias de la

empresa, organización que lo utilice, clasificadas en función de su estado: nuevo, abierto, en curso, cerrado o anulado. El sistema se utiliza para brindar soporte técnico a usuarios, gestión de recursos, personal y atención al cliente. Se desarrolla con tecnología ASP.NET, es parte de una integración de programas de gestión (Artologik, 2009). Algunas de sus ventajas son:

- Adaptable a cada sector y proyecto.
- Información centralizada
- Recordatorios y alerta web
- Base de conocimiento donde todas las incidencias registradas en HelpDesk quedan almacenadas y disponibles para su consulta por el equipo de soporte.

Pero su adopción no es posible ya que casi toda su información se encuentra en la nube lo que hace obligatorio el uso de internet además de que para su utilización hay que efectuar pagos mensuales (Artologik, 2009).

#### **Sistema de Información Científica de Andalucía. SICA**

Es una aplicación de la Secretaría General de Universidades, Investigación y Tecnología adscrito a la Agencia Andaluza de Evaluación de la Calidad y la Acreditación Universitaria, cuyo objetivo principal es recopilar y ofrecer información sobre la actividad científica e investigadora de las Universidades Andaluzas. Esta aplicación que sirve de apoyo curricular para los investigadores de las diferentes universidades de Sevilla (Andalucía, 2018).

El SICA ha tenido un claro compromiso a favor del Software Libre, ayudando a su difusión, su distribución, su conocimiento y especialmente, incorporándolo como parte de su propia infraestructura TIC.

El SICA es uno de los recursos más solicitados durante los períodos puntuales de convocatorias abiertas a investigadores esto se evidencia en la gráfica mostrada a continuación:



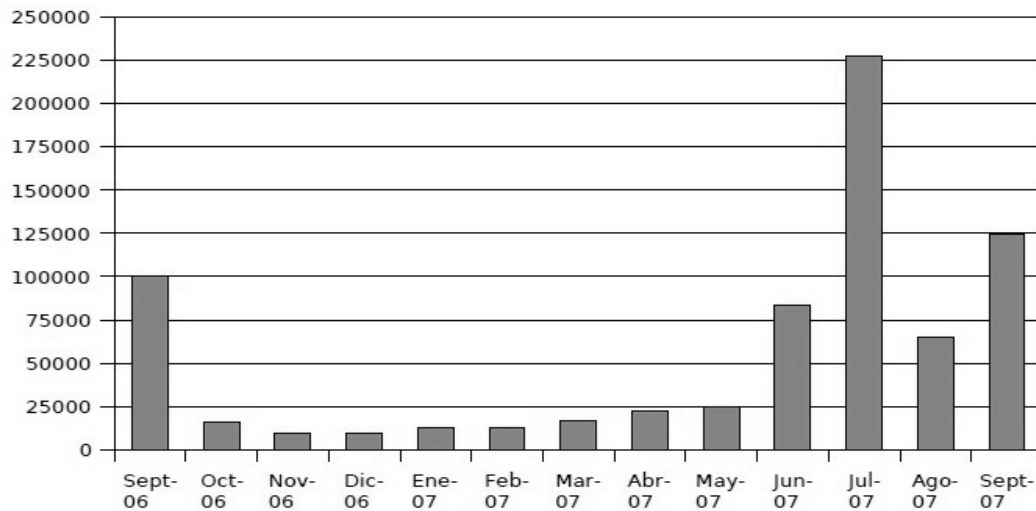


Figura 1. Número total de accesos a SICA (Andalucía, 2018).

Es un sistema de ámbito regional que agrupa la producción científica y que a medida que esta se genera, es validada en poco tiempo. Facilitando un análisis fiable y evolutivo de las políticas científicas (Andalucía, 2018).

### **Índice de Revistas de Educación Superior e Investigación Educativa. IRESIE**

Es un banco de información sobre la educación. El IRESIE es un sistema que detecta, selecciona, analiza, sistematiza, procesa en forma automatizada y difunde la información que sobre educación superior e investigación educativa se publica en revistas mexicanas y extranjeras y facilita el flujo de información entre el productor y el usuario (Verdugo, 1981).

Tiene como objetivo apoyar el desarrollo de la investigación, planeación y administración de la educación. Utiliza el formato *html*. Propicia la actualización permanente de investigadores, docentes, profesionales, estudiantes y del usuario en general. Presenta un cúmulo de referencias bibliográficas muy extenso, lo que posibilita conocer una vasta gama de experiencia e innovaciones educativas de diferentes sociedades y principalmente tener una imagen objetiva de la realidad educativa nacional e internacional (Verdugo, 1981). Una de las desventajas de este sistema es el costo de la suscripción al mismo y la limitada difusión de las publicaciones.

## **1.2.2 Sistemas de gestión de la información en el ámbito nacional**

### **Sistema de Información para la Gestión de Ciencia, Innovación y Tecnología en las Facultades de Ciencias Médicas. SINFORCIT**

Es una aplicación web como herramienta informática y componente del Sistema de Información al usuario, que facilita el servicio informativo existente para los usuarios de los departamentos de investigaciones de las facultades de Ciencias Médicas. Basada en multimedia e hipertexto, para publicar determinada información en las redes. Inicialmente se compone de protocolo *HTTP* y del lenguaje marcado de hipertexto (*HTML* por sus siglas en inglés) (Alpízar, 2015).

Esta herramienta sin costos resulta factible de colocar en una Red de servicios Web o de llevar el usuario en soporte portátil con la información necesaria para cumplir sus demandas. Está dirigido al uso de profesores, departamentos docentes de cualquier nivel de salud, pues los orienta en el “saber qué” y el “cómo” brindar la información (Alpízar, 2015).

Algunas de las desventajas de SINFORCIT son: se encuentra que la gestión de los procesos no siempre se conoce por los profesionales que asumen la conducción o ejecución de los procedimientos. Esta situación pone en riesgo el cumplimiento de los procesos del sistema en cuestión. Además, el movimiento de los cuadros que atienden esta esfera, fundamentalmente en las instituciones de la base, repercute en el desconocimiento de las funciones. Aún se encuentra en proceso de generalización y no cumple completamente con la optimización de recursos (Alpízar, 2015).

### **Sistema de Gestión de Investigación de la Facultad 2 de la Universidad de la Ciencias Informáticas**

El sistema tiene como objetivo principal automatizar los procesos de gestión de la información asociada a la investigación y el postgrado en la Facultad 2, lo que permite una mayor rapidez en el manejo de la información referente a ambos procesos. Dentro de las funcionalidades de este sistema están: gestionar proyectos de investigación, cursos de postgrado, líneas de investigación, generar reportes, gestionar grupos de investigación, líneas temáticas, realizar solicitud de cursos de postgrado y realizar solicitud de programas de postgrado. Utiliza el lenguaje marcado de hipertexto (*HTML* por sus siglas en inglés) y *PHP* como lenguaje de programación.

### **1.2.3 Resultados del estudio de los sistemas homólogos**

El estudio de los sistemas de gestión de información en el ámbito nacional e internacional permitió realizar un resumen donde se analizaron los siguientes aspectos: Las herramientas internacionales estudiadas son sistemas privativos, utilizar las APIs que brindan crean una dependencia tecnológica y no cumple con los objetivos definidos al requerir el uso de internet. Las herramientas nacionales son sistemas informativos., los cuales no están concebidos para la gestión de la información relacionada con las problemáticas de carácter científico de una institución, por lo que no cumplen con lo que se quiere hacer.

Muchos de estos sistemas no satisfacen las necesidades, pues no tratan la gestión de contenido científico de la manera en que se necesita. Uno de los sistemas analizados es el IRESIE que es una herramienta de búsqueda por lo que no se puede adoptar su solución al no cumplir con los requisitos necesarios para ser un sistema de gestión. Sin embargo, el análisis de los mismos permitió identificar funcionalidades y tecnologías que pueden contribuir a la implementación de la propuesta de solución. El sitio web HelpDesk plantea una solución similar a la que se quiere obtener pero es un software privado por lo que no es factible su uso, solo se contará con el marco teórico que plantea. Está además el Sistema de Gestión de Investigación de la Facultad 2 que plantea otra de las soluciones, pero está enfocado solamente en la gestión de la información asociada al postgrado lo cual hace que no se pueda adoptar pues se desea gestionar la información asociada al pregrado también, sin embargo, se identificó y utilizó la funcionalidad generar reportes y gestionar líneas de investigación de este sistema.

## **1.3 Descripción de los lenguajes, herramientas y tecnologías utilizadas**

Por ser la propuesta de solución un sistema para la gestión del banco de problemas científicos de la facultad se realiza un estudio y análisis de las herramientas y tecnologías existentes y de otras relacionadas con el desarrollo de los SG. Los elementos planteados facilitan la futura integración y compatibilidad con el sistema.

### **1.3.1 Marco de trabajo**

Un marco de trabajo proporciona estructura al código fuente, lo que permite al desarrollador a crear código legible y fácil de mantener. Facilita la programación de aplicaciones y encapsula operaciones complejas en instrucciones sencillas. Uno de los desafíos difíciles de los proyectos de desarrollo es crear sitios web y aplicaciones de calidad en el menor tiempo posible (2017).

## **Symfony**

Symfony es actualmente el más escalable marco de trabajo de *PHP* (del inglés *Hipertext Pre-procesor*). Es un marco de trabajo basado en el patrón Modelo Vista Controlador (MVC) para el desarrollo de aplicaciones web. Se pueden crear aplicaciones y sitios web rápidos y seguros de una forma profesional (Coders, 2017).

Proporciona la ventaja de otros componentes como plantillas, traducción, validador, configuración de formulario, lo que lo convierte en uno de los mejores marcos de PHP (Coders, 2017).

Los principales beneficios comerciales de Symfony son (Coders, 2017):

- Desarrollo rápido
- Soporte a largo plazo
- Mantenibilidad fácil
- Proporciona seguridad y flexibilidad
- Fiabilidad probada

## **Laravel**

Laravel es un marco de trabajo de código abierto para desarrollar aplicaciones y servicios web con PHP 5 y PHP 7, tiene como objetivo permitir el uso de una sintaxis elegante y expresiva para crear código de forma sencilla y permitiendo multitud de funcionalidades. Este es uno de los marcos de trabajo de desarrollos web más importantes que viene con un enrutamiento reparador (Coders, 2017).

El marco de trabajo también incluye PHP nativo. Proporciona una base sólida y una base para el desarrollo de aplicaciones web. Algunos de sus beneficios comerciales se mencionan a continuación (Coders, 2017):

- Migración de datos sin problemas
- Seguridad de primera clase
- Integración robusta de herramientas
- Baja sobrecarga del servidor
- Gran comunidad de desarrolladores

La principal desventaja que tiene es su fuerte orientación hacia la programación estática, usa una abusiva cantidad de clases y métodos estáticos. Además la curva de aprendizaje es compleja si no hay un buen

manejo del lenguaje. Su rendimiento no es mejor que el de Symfony. Laravel usa algunos componentes robustos y estables que pertenecen a Symfony, esto significa que Laravel básicamente nace de Symfony.

Por lo anteriormente demostrado se ha seleccionado el framework Symfony en su versión 3.1.0 porque entre las ventajas clave de las características de desarrollo de este marco se encuentra la alta escalabilidad que posee, es más rápido que muchos otros marcos de trabajo de PHP si se usa adecuadamente, es adaptable a las necesidades del usuario, ofrece un rendimiento óptimo y proporciona un componente reutilizable que reduce los costos generales. Cumpliendo así con las necesidades del sistema a desarrollar.

### **1.3.2 Servidor de Base de Datos**

Una de las tareas más comunes de las aplicaciones web dinámicas es la persistencia y lectura de la información en las bases de datos. *Symfony* se integra a *Doctrine*, biblioteca que se utiliza para simplificar las operaciones que se realizan en las bases de datos, posee controladores que permiten la compatibilidad con servidores de código abierto como es el caso de *MySQL* (Pacheco, 2013).

#### **MySQL**

Es un SGBD relacional, su diseño multihilo le permite soportar una gran carga de datos de forma eficiente. *MySQL* es el gestor de base de datos de código abierto más utilizado a nivel mundial por su rapidez y facilidad de uso datos ofrecidos por Oracle (2017). Su utilización está enfocada en aplicaciones web dinámicas escritas en *PHP* por la optimización de consultas sencillas y su compatibilidad con el servidor web *Apache*. Este se caracteriza por su nivel de seguridad, es multiplataforma, admite hasta 32 índices por tablas, la variedad de tipos de datos para las columnas y la utilización de *triggers* (Oracle, 2017).

Se selecciona *MySQL* versión 5.7 como SGBD por su velocidad de procesamiento en operaciones y garantizar un mayor rendimiento cuando se consultan las bases de datos, su bajo consumo ante servidores con bajas prestaciones y su facilidad de uso en la administración.

### **1.3.3 Servidor web**

Un servidor web es un programa que utiliza el protocolo de transferencia de hipertexto *HTTP* (Hypertext Transfer Protocol), para servir los archivos que forman páginas web a los usuarios, en respuesta a sus solicitudes, que son reenviados por los clientes *HTTP* de sus computadoras. Las computadoras y los dispositivos dedicados también pueden denominarse Servidores Web (Rouse, 2016).

**Apache v2.4** es un software de servidor web gratuito y de código abierto con el cual se ejecutan el 46% de los sitios web de todo el mundo. El nombre oficial es *Apache HTTP Server*, y es mantenido y desarrollado por la *Apache Software Foundation* (Bustos, 2018).

Apache es altamente personalizable y tiene una estructura basada en módulos. Los módulos permiten a los administradores del servidor activar y desactivar funcionalidades adicionales. Apache tiene módulos de seguridad, almacenamiento en caché, reescritura de URL, autenticación de contraseña (Bustos, 2018).

### **1.3.4 Lenguaje y Herramienta de Modelado**

El lenguaje de modelado es un lenguaje informático gráfico o textual que sigue un conjunto semántico de reglas y marcos para el diseño y construcción de estructuras y modelos. Se utiliza el Lenguaje Unificado de Modelado (UML por sus siglas en inglés) en su versión 2.1, diseñado para visualizar, especificar, construir y documentar software orientados a objetos (2017).

Las herramientas de Ingeniería de Software Asistida por Computadora (por sus siglas en inglés CASE) son una de las principales herramientas para el modelado. Estas herramientas, brindan soporte a todas las actividades del proceso propuesto por la Ingeniería del Software y a todas aquellas actividades transversales que se aplican a lo largo de todo el proceso, desde la gestión del proyecto hasta la gestión del producto. Posibilitan al ingeniero automatizar actividades manuales y mejorar la visión general de la ingeniería (2013)

Visual Paradigm para UML es una herramienta para el modelado de aplicaciones, ideal para ingenieros de software, analistas de sistemas y arquitectos de sistemas, que están interesados en la construcción de sistemas a gran escala y necesitan confiabilidad y estabilidad en el desarrollo orientado a objetos. Permite la generación de bases de datos, conversión de diagramas entidad-relación a tablas de base de datos, mapeos de objetos y relaciones, ingeniería directa e inversa, la gestión de requisitos de software y la modelación de procesos del negocio (2016). Para la presente investigación, se utiliza Visual Paradigm en su versión 8.0

### **1.3.5 Lenguajes de programación**

Los lenguajes de programación pueden utilizarse para crear programas que controlen el comportamiento físico y lógico de una máquina. Son un conjunto de reglas semánticas y sintácticas que se utilizan para la codificación de instrucciones de un programa o algoritmo de programación (2017). Para el desarrollo de la

propuesta de solución los lenguajes de programación se dividen en dos grupos: del lado del cliente y del lado del servidor.

### 1.3.6 Lenguaje de programación del lado del servidor

Un lenguaje de programación es un conjunto de reglas semánticas y sintácticas que se utilizan para la codificación de instrucciones de un programa o algoritmo de programación (Álvarez, 2016). Para el desarrollo del módulo se emplea el lenguaje *PHP* en su versión 7.0, es un lenguaje de programación gratuito e independiente de plataforma, rápido, con una gran librería de funciones y un extenso volumen de documentación (PHP, 2017). Según las investigaciones de (Martínez, 2014) se puede afirmar que las principales ventajas de *PHP* son:

- Aplicación de técnicas de la programación orientada a objeto para el desarrollo de aplicaciones web dinámicas.
- Capacidad de expandir su potencial utilizando módulos y extensiones.
- Simplificación de distintas especificaciones, como es el caso de la definición de las variables primitivas.
- Posee una amplia documentación en varios idiomas en su sitio web oficial, entre la cual se destaca que todas las funciones del sistema están explicadas y ejemplificadas en un único archivo de ayuda.
- Tiene manejo de excepciones.
- Capacidad de conexión con la mayoría de los gestores de base de datos que se utilizan en la actualidad, destaca su conectividad con MySQL y PostgreSQL.

### 1.3.7 Lenguaje de programación del lado del cliente

Los lenguajes de programación del lado del cliente se utilizan para su integración en páginas web. La base de la programación del lado cliente consiste en escribir un *script* y hacer que este funcione en el navegador sin tener que viajar hasta el servidor para ejecutarlo (Jimenez, 2015).

**JavaScript v.1.8.5:** Es un lenguaje de programación interpretado, por lo que no es necesario compilar los programas para ejecutarlos. Se utiliza parcialmente para crear páginas web dinámicas y trabaja sobre un enfoque útil y práctico. Los programas escritos en JavaScript pueden ser probados en cualquier navegador web, sin necesidad de procesos intermedios (Bustos, 2018).

Se utiliza el Lenguaje de Marcado de Hipertexto (*HTML* por sus siglas en inglés) en su versión 5.0. *HTML* se utiliza en la traducción y descripción de la estructura y la información en forma de texto, así como para complementar el texto con objetos e imágenes (2018). Es un lenguaje básico predefinido en el marco de trabajo seleccionado.

Para definir la apariencia o estilo de las páginas web escritas en *HTML* del sistema propuesto se utiliza *CSS* versión 3.0. *CSS* permite separar el contenido de la forma y proporciona a los diseñadores el mantenimiento de un control más preciso sobre la apariencia de las páginas web. Su novedad más importante consiste en la incorporación de mecanismos para mantener un mayor control sobre el estilo de los elementos que se muestran en la página (Arume, 2015).

### **1.3.8 Entorno Integrado de Desarrollo**

**NetBeans IDE v8.2:** Es un entorno de desarrollo visual de código abierto que inicialmente solo servía para aplicaciones programadas mediante Java. A partir de la versión 6, admite varios lenguajes de programación, ya sea a través del soporte integrado o instalando complementos adicionales. Los lenguajes de programación soportados nativamente por NetBeans son: Java, C, C++, PHP, HTML y JavaScript, Groovy, Scala y otros son compatibles a través de complementos adicionales. Con NetBeans es posible diseñar aplicaciones con solo arrastrar y soltar objetos sobre la interfaz de un formulario. Posibilita la elaboración de potentes aplicaciones para el escritorio, web y dispositivos portátiles, como teléfonos inteligentes o tabletas, sin que cambie la forma de programar. La programación se realiza a través de componentes de software modulares, también llamados módulos. Incluye un avanzado editor para varios lenguajes, editor de perfiles y un detector de errores, además de herramientas para el control de versiones y el desarrollo colaborativo. Proporciona módulos a través de su página web, que se pueden integrar en él para conseguir mejores aplicaciones (López, 2012).

## **1.4 Metodología de desarrollo**

Una metodología de desarrollo de software es un marco de trabajo que se usa para estructurar, planificar y controlar el proceso de desarrollo de un software. Actualmente, las metodologías de desarrollo de software pueden considerarse como una base necesaria para la ejecución de cualquier proyecto serio. Estas metodologías son necesarias para poder realizar un proyecto profesional, tanto para poder desarrollar efectiva y eficientemente el software, como para que sirva de documentación y se puedan



rendir cuentas de los resultados obtenidos (Sánchez, 2015). En la actualidad existen varias metodologías de desarrollo de software, y cada una de estas debe ser adaptada a las características de cada proyecto. Para el desarrollo de la solución se empleará AUP-UCI.

## **AUP-UCI**

Es una variación de la metodología de desarrollo AUP (Proceso Unificado Ágil) creada por la UCI. Fue creada con el objetivo de ser adaptada al ciclo de vida de los proyectos de la universidad. Esta metodología describe de una manera simple y fácil de entender la forma de desarrollar aplicaciones de software de negocio usando técnicas ágiles. Se definen 3 fases para componer el ciclo de vida de un proyecto:

**Inicio:** realiza un estudio inicial de la organización cliente que permite obtener información fundamental acerca del alcance del proyecto, realizar estimaciones de tiempo, esfuerzo, costo y decidir si se ejecuta o no el proyecto.

**Ejecución:** ejecuta las actividades requeridas para desarrollar el software, incluyendo el ajuste de los planes del proyecto considerando los requisitos y la arquitectura. Durante el desarrollo se modela el negocio, obtiene los requisitos, se elabora la arquitectura y el diseño, se implementa y se libera el producto.

**Cierre:** analiza tanto los resultados del proyecto como su ejecución y se realizan las actividades formales del cierre del proyecto.

## **1.5 Herramientas de validación**

### **JMeter**

La aplicación JMeter de Apache es un software de código abierto, desarrollada en Java y diseñada para cargar el comportamiento funcional de la prueba y medir el rendimiento. Originalmente, fue diseñado para probar aplicaciones web, pero desde entonces se ha expandido a otras funciones de prueba. JMeter se puede usar para probar el rendimiento tanto en aplicaciones web dinámicas como estáticas. Además, se puede emplear para simular una gran carga en un servidor, grupo de servidores red u objeto para probar su fortaleza o analizar el rendimiento general bajo diferentes tipos de carga. Las características de Apache JMeter incluyen: capacidad para cargar y probar el rendimiento de diferentes aplicaciones, servidores y

protocolos. Fácil correlación mediante la capacidad de extraer datos de los formatos de respuestas más populares, HTML, JSON, XML o cualquier formato de texto, núcleo altamente extensible, almacenamiento en caché y portabilidad completa (ApacheJMeter, 2017).

### **Acunetix**

Acunetix WVS, por sus siglas en inglés (Web Vulnerability Scanner) es una herramienta de seguridad de aplicaciones web automatizada. Es capaz de escanear cualquier sitio o aplicación web que es accesible a través del protocolo HTTP/HTTPS. Sin embargo, no todas las pruebas se pueden realizar de forma automática, por lo tanto, Acunetix WVS proporciona herramientas de penetración manuales para pruebas particulares. Comprueba diferentes vulnerabilidades (por ejemplo, inyección de SQL, Cross Site Scripting). Hasta la fecha Acunetix comprueba sobre más de 500 tipos diferentes de vulnerabilidades (Acunetix, 2012).

## **1.6 Conclusiones del capítulo**

En este capítulo se realizó un profundo estudio sobre los conceptos relacionados con los sistemas de gestión de información. También se realizó un análisis de soluciones existentes a nivel internacional y nacional para analizar la forma en que gestionan la información científica, pero no se encontró ninguno que cumpla con el proceso de gestión de los problemas de investigación, pero si se obtuvieron características y funcionalidades importantes para el desarrollo del sistema como gestionar líneas de investigación y generar reportes. Además se realizó un análisis completo de la metodología a utilizar a lo largo del sistema propuesto, en este caso AUP-UCI ya que se adapta al ciclo de vida de los proyectos de la universidad. Las herramientas seleccionadas para el desarrollo del sistema fueron: como entorno de desarrollo integrado el Netbeans, como gestor de base de datos MySQL, como servidor web Apache, como marco de trabajo Symfony y el Visual Paradigm para el modelado, debido a que soporta el ciclo de vida completo del desarrollo de software.

## Capítulo 2: Análisis y diseño del sistema para la gestión del banco de problemas científicos de la Facultad 1.

En el presente capítulo se describen las principales características de la propuesta de solución y se identifican los requisitos funcionales y no funcionales. Se describe la arquitectura de software, los patrones de diseños y los artefactos generados para darle solución al problema planteado en la investigación.

### 2.1 Modelo Conceptual

Un modelo conceptual es un artefacto de la disciplina de análisis, construido con las reglas UML. Tiene como objetivo comprender y describir las clases más importantes, así como, identificar y explicar los conceptos significativos en el dominio del problema, identificando los atributos y las asociaciones existentes entre ellos (Sommerville, 2011).

- **Persona:** Persona encargada de crear los problemas de investigación puede estar vinculada a un centro o no.
- **Centro:** Centro de producción al que puede estar vinculada una persona.
- **Facultad:** A donde pertenecen las personas que no están vinculadas a un centro.
- **Problemas de Investigación:** Propuestos por la persona y es lo principal del proceso.
- **Comisión Científica del Centro:** Es la encargada de revisar los problemas de investigación propuestos por la persona que pertenece al centro.
- **Jefe de la Comisión:** Es el encargado de asignar los problemas de investigación aprobados a las personas del centro.
- **Comisión Científica de la Facultad:** Es la encargada de aprobar los problemas de investigación propuestos según estén bien formulados.
- **Comité de Tesis:** Decide a que persona de la facultad se le asigna el problema de investigación.

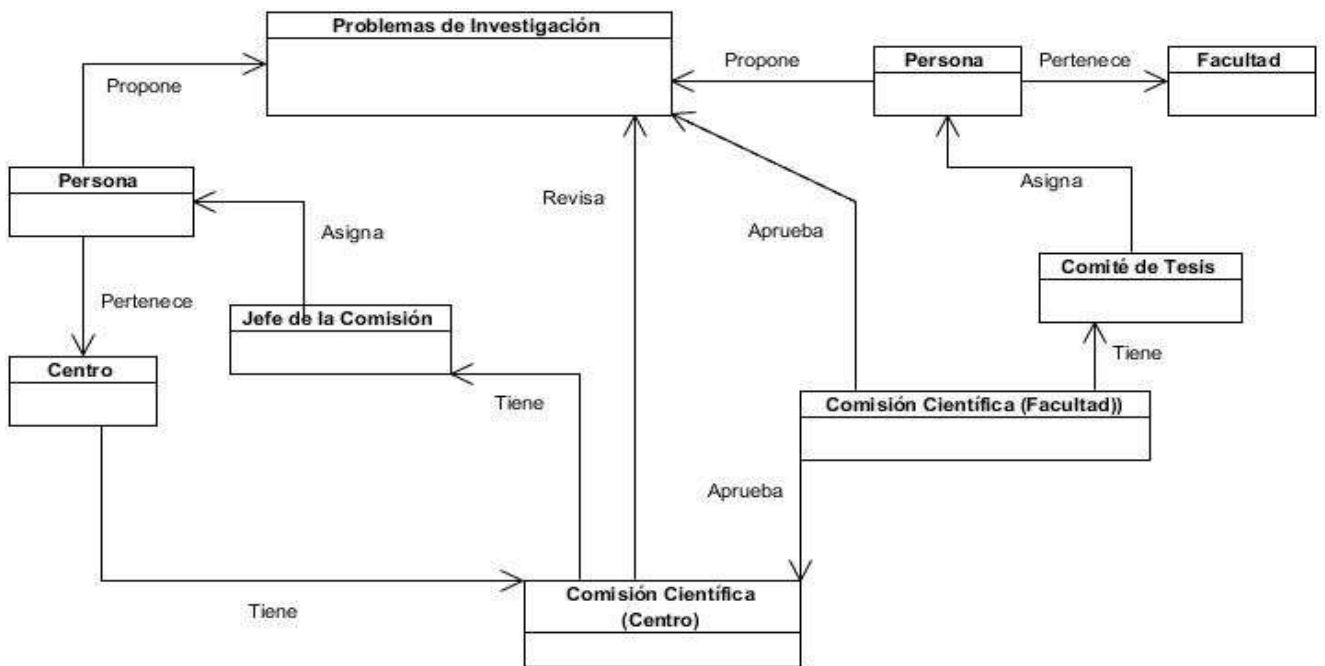


Figura 3. Modelo Conceptual (Elaboración propia)

## 2.2 Descripción de la propuesta de solución

El sistema para la gestión del banco de problemas científicos brindará la posibilidad de gestionar las problemáticas que sean originadas en la facultad por cualquier usuario del sistema, las cuales podrán ser revisadas y aprobadas o no por un grupo de expertos, en este caso la comisión científica de la facultad. El mismo tiene como objetivo organizar el sistema de Ciencia Tecnología e Innovación de la Facultad 1, a partir de la gestión de un banco de problemas que responde a las necesidades de la facultad y en concordancia con las líneas de investigación de la universidad. La principal característica de la propuesta de solución es que mejora el proceso de asignación de tesis. El sistema además tendrá un banco de resultados donde se mostrarán los problemas de investigación resueltos y clasificados según el nivel de pregrado, maestría y doctorado, lo que contribuye a concretar la pirámide de investigación.

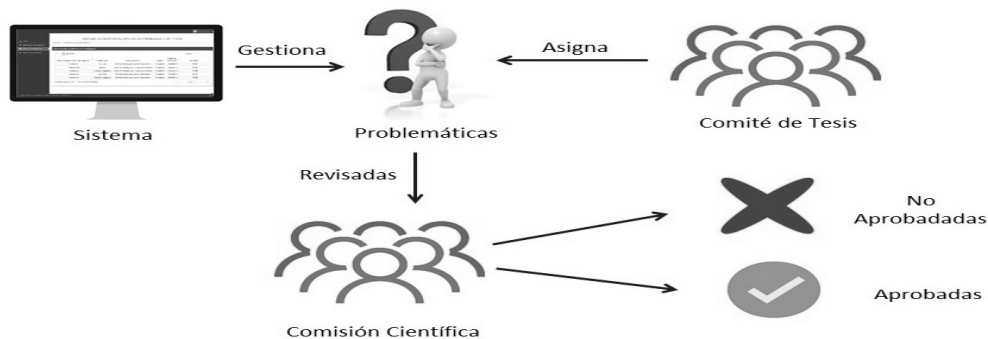


Figura 2. Propuesta de solución (Elaboración propia)

## 2.3 Requisitos de la propuesta de solución

Los requisitos se pueden clasificar en funcionales y no funcionales. Pueden documentar las interfaces externas, describir la funcionalidad y el rendimiento del sistema (Sommerville, 2011). En esta sección se describen los servicios que se proponen al usuario.

### 2.3.1 Requisitos Funcionales

Los requisitos funcionales (RF) describen lo que el sistema debe hacer. Estos dependen del tipo de software que se desarrolle, los posibles usuarios y el enfoque tomado por el proyecto. Además describen con detalle la función del sistema, sus entradas, excepciones y salidas (Sommerville, 2011).

En la Tabla 1 se muestran los requisitos funcionales de la propuesta de solución, a los cuales se les asignó una prioridad teniendo en cuenta la importancia fijada por el cliente a partir de sus necesidades.

Tabla 1. Requisitos Funcionales. (Elaboración propia)

Nº	Nombre	Descripción	Prioridad	Complejidad
RF1	Autenticar usuario mediante LDAP	El sistema debe permitir autenticar un usuario mediante LDAP	Media	Baja
RF2	Gestionar problema de investigación	El sistema debe permitir gestionar los problemas de investigación	Alta	Media

RF3	Buscar problema de investigación	El sistema debe permitir crear una línea científica	Baja	Baja
RF4	Gestionar línea científica	El sistema debe permitir gestionar las líneas científicas	Baja	Baja
RF5	Restaurar base de datos	El sistema debe permitir restaurar la base de datos	Alta	Media
RF6	Salvar base de datos	El sistema debe permitir salvar la base de datos	Alta	Media
RF7	Generar trazas por IP	El sistema debe permitir generar trazas por IP	Media	Media
RF8	Generar trazas por fecha	El sistema debe permitir generar trazas por fecha	Media	Media
RF9	Generar trazas por usuario	El sistema debe permitir generar trazas por usuario	Media	Media
RF10	Generar reportes	El sistema debe permitir generar reportes sobre los problemas de investigación	Media	Media
RF11	Aprobar problema de investigación	El sistema debe permitir aprobar un problema de investigación	Alta	Media
RF12	Asignar problema de investigación	El sistema debe permitir asignar un problema de investigación	Alta	Media
RF13	Insertar resumen del problema de investigación resuelto	El sistema debe permitir insertar el resumen de un problema de investigación resuelto	Media	Media
RF14	Mostrar resumen del problema de investigación resuelto	El sistema debe permitir mostrar el resumen de un problema de investigación resuelto	Media	Media
RF15	Exportar reportes a pdf	El sistema debe permitir exportar a pdf todos los reportes	Media	Baja

### 2.3.2 Requisitos No Funcionales

Los requisitos no funcionales (RnF) hacen referencia a las propiedades emergentes del sistema: fiabilidad, el tiempo de respuesta y la capacidad de almacenamiento, son limitaciones sobre servicios o funciones que ofrece el mismo. Los requisitos no funcionales se aplican al sistema como un todo, más que a características o a servicios individuales (Sommerville, 2011).

#### RnF1. Usabilidad

**RnF1.1** Debe ser intuitivo, los requisitos funcionales son agrupados por responsabilidad, para permitir a los administradores de red de poca experiencia, hacer uso del mismo.

**RnF1.2** El sistema podrá ser utilizado por personas que tengan un conocimiento básico en el manejo de las computadoras.

### **RnF3. Hardware**

**RnF3.1** El servidor donde se instale el sistema debe tener una memoria *RAM* de 1GB o superior y un procesador a una velocidad 2.10 GHz o superior y para el servidor de base datos 10 GB de disco duro.

### **RnF4. Software**

**RnF4.1** Los dispositivos de los usuarios deben contar con navegadores web que soporten HTML5 y CCS3.

**RnF4.1** Se requiere la instalación de un servidor web (recomendándose Apache en su versión 2.4 o superior) y PHP v7 o superior para el correcto funcionamiento de la aplicación web.

### **RnF5. Seguridad**

**RnF5.1** La información manejada en el sistema debe estar protegida del acceso no autorizado mediante los mecanismos de control de acceso que establece el marco de trabajo Symfony.

### **RnF6. Mantenimiento y Soporte**

**RnF6.1** El sistema debe ser escalable, permitiendo incorporarle nuevas funcionalidades sin afectar las existentes.

**RnF6.2** Fácil Instalación.

## **2.4 Arquitectura de Software**

La arquitectura de software es un conjunto de patrones que definen la organización de un sistema, sus componentes, el ambiente, y los principios que orientan su diseño y evolución. Los patrones arquitectónicos son la estructura del sistema que comprende a los elementos de software, las propiedades visibles externamente de dichos elementos y las relaciones entre ellos (Rodríguez y otros, 2016). Para el sistema se escogió la arquitectura Modelo-Vista-Controlador explicada a continuación:

### **Modelo – Vista – Controlador**

Symfony basa su flujo de trabajo en el patrón arquitectónico Modelo – Vista – Controlador, implementada de forma que el desarrollo de aplicaciones sea rápido y sencillo. Este patrón está formado por tres niveles:

- **Modelo:** Representa la información con la que trabaja el sistema, su lógica de negocio. El modelo

está representado en los archivos del directorio: *Entity*.

- **Vista:** Transforma el modelo en una página web y permite al usuario interactuar con este. Las vistas del módulo están agrupadas en el directorio: *app/Resources/view*.
- **Controlador:** Se encarga de procesar las interacciones del usuario, y realiza los cambios apropiados en el modelo o la vista. Las clases controladoras de la propuesta de solución están situadas en el directorio: *Controller*.



Figura 4. Patrón M-V-C (Elaboración propia)

En el controlador se encuentran las acciones que permiten gestionar los problemas de investigación, es el núcleo del sistema y contiene la lógica de la aplicación. Estas acciones utilizan el modelo y envían las variables a la vista, que es la encargada de originar las páginas que son mostradas como resultado de las acciones. En el modelo se encuentra la clase relacionada con los problemas de investigación para el acceso y la manipulación de los datos, realizada mediante objetos. El controlador es la capa intermediaria entre la Vista y el Modelo.

## 2.5 Patrones de Diseño

Los patrones de diseño son una técnica para resolver problemas comunes en el desarrollo web y otros ámbitos referentes al diseño de interacción o interfaces. Establece una relación entre un determinado contexto, un problema y una solución. Para el diseño de software, el contexto permite al lector entender el ambiente en el que reside el problema y qué solución sería apropiada en dicho ambiente. Un conjunto de requerimientos, incluidas limitaciones y restricciones, actúan como sistema de fuerzas que influyen en la manera en la que puede interpretarse el problema en este contexto y en cómo podría aplicarse con eficacia la solución (Pressman, 2010).



### 2.5.1 Patrones Generales de Software para la Asignación de Responsabilidades (GRASP)

Los patrones GRASP, fomentan una serie de buenas prácticas para el diseño de software (Cárdenas, 2016). De esta familia, para el desarrollo del sistema se utilizaron los siguientes patrones (Hamon, 2014):

**Experto:** Es uno de los patrones que más se utiliza cuando se trabaja con Symfony. Asigna las responsabilidades a aquellos objetos que disponen de la información para hacerlo. En el sistema para la gestión del banco de problemas científicos se evidencia en la clase entidad *ProblemaInvestigacion.php*, encargada de obtener la información de los problemas almacenados en la base de datos.

**Alta Cohesión:** La cohesión es una medida de cuán relacionadas y enfocadas están las responsabilidades de una clase. Las mismas poseen un número relativamente pequeño de responsabilidades, las clases realizan solo las funciones para las cuales fueron creadas, esto conlleva a un bajo acoplamiento y fomenta la reutilización. En el sistema, este patrón se manifiesta en la clase controladora *ProblemaInvestigacionController*, colabora y delega responsabilidades a la clase *ClasificacionType* para la creación de formularios relacionados con la entidad *Clasificacion*.

**Controlador:** Un Controlador es un objeto de interfaz no destinado al usuario, se encarga de manejar un evento del sistema a una clase que represente el sistema global. En el sistema, los problemas generados por el usuario son redirigidos a una clase controladora que realiza las operaciones solicitadas.

**Bajo Acoplamiento:** Es una medida de la fuerza en las conexiones de una clase con otras, la recurrencia y las conexiones a ella. Una clase con bajo acoplamiento no depende de muchas clases. Su utilización se evidencia en que las clases controladoras del sistema no se relacionan entre sí, lo que disminuye las dependencias entre las mismas.

### 2.5.2 Patrones Gang of Four (GOF)

Describen las formas comunes en que diferentes tipos de objetos, pueden ser organizados para trabajar unos con otros. Gestionan la relación entre clases y la formación de estructuras de mayor complejidad. Además, permiten crear grupos de objetos para ayudar a realizar tareas complejas (Craig, 2003). Los patrones GOF utilizados en el sistema se relacionan a continuación:

### Estructurales:

**Decorator (Decorador):** Permite añadir responsabilidades adicionales a un objeto dinámicamente para proporcionar una alternativa flexible a la especialización mediante herencia si se añaden nuevas funcionalidades. En el sistema se observa este patrón en la vista **base.html.twig** de ella heredan el resto de las páginas de la aplicación. Esta vista contiene los elementos que son comunes para el diseño y estructura del sistema.

### Otros:

**Registry:** Este patrón es útil para los desarrolladores en la Programación Orientada a Objetos. Es un medio sencillo y eficiente de compartir datos y objetos en la aplicación, sin la necesidad de conservar numerosos parámetros o utilizar variables globales. Se aplica en la clase de configuración (**config.yml**) que es la encargada de guardar las variables globales del módulo, como *drivers* de conexión a la base de datos y rutas bases del proyecto.

## 2.6 Historias de usuario

La metodología AUP-UCI, en su escenario 4 para la disciplina Requisitos, genera como artefacto las Historias de Usuario (HU) (Sánchez, 2015), estas se utilizan para especificar los requisitos de software en las aplicaciones. Las HU se descomponen en tareas de programación y describen las características que el sistema debe cumplir, están escritas en un formato legible por el cliente, sin necesidad de sintaxis técnicas (Ordoñez y otros, 2015).

A continuación, se muestran dos (2) HU de las quince (15) que fueron generadas, perteneciente al RF 2 y el RF 5, el resto puede ser consultada en el Anexo 1 de la presente investigación.

Tabla 2. HU\_2 Gestionar Problema de Investigación (Elaboración propia)

Historia de Usuario	
<b>Número:</b> HU_2	<b>Nombre:</b> Gestionar Problema de Investigación
<b>Programador responsable:</b> Ana Carla Del Real Baró	<b>Iteración asignada:</b> 1
<b>Prioridad:</b> Alta	

**Descripción:** Los usuarios pueden crear, editar, mostrar o eliminar un problema de investigación después de haberse logueado en el sistema.

**Observaciones:** Cualquier usuario puede gestionar un problema de investigación, solo debe rellenar los campos requeridos correctamente y el sistema le mostrará un mensaje, dependiendo de que acción realice, un mensaje de que se realizó la acción correctamente y un mensaje de error en caso contrario.

**Prototipo de Interfaz:**

Banco de Problemas

Título

Clasificación

Línea Científica

Descripción

Nombre

Apellidos

Centro

Tabla 3. HU\_5 Restaurar Base de Datos (Elaboración propia)

Historia de Usuario	
<b>Número:</b> HU_5	<b>Nombre:</b> Restaurar Base de Datos
<b>Programador responsable:</b> Ana Carla del Real Baró	<b>Iteración asignada:</b> 1
<b>Prioridad:</b> Media	
<b>Descripción:</b> El sistema debe permitir restaurar la base de datos	

**Observaciones:** Al restaurar la base de datos se debe tener un servidor ip, nombre de usuario, contraseña y la fecha que se desea restaurar. El administrador será el encargado de restaurar la base de datos.

**Prototipo de interfaz:**

Restaurar Base Datos

Servidor IP

Fecha

Nombre de Usuario

Contraseña

## 2.7 Diagrama de clases de diseño

Un Diagrama de Clases del Diseño (DCD) muestra la especificación de las clases de una aplicación, sus asociaciones, atributos y métodos, interfaces, navegabilidad y dependencias. Las clases de diseño de los DCD definen entidades y no conceptos del mundo real (UNAD, 2016). Para la presente investigación se generaron un total de cinco (5) DCD relacionados con los requisitos de prioridad alta, de ellos se exponen dos (2) correspondiente a la HU\_2 y a la HU\_2, el resto de los DCD puede ser consultado en el Anexo 2.

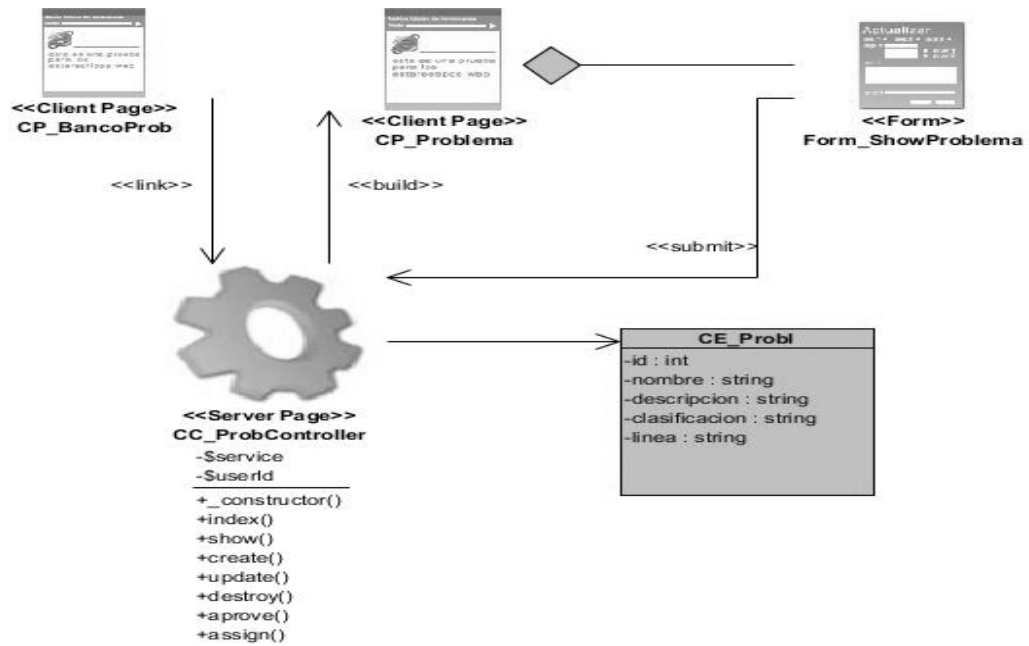


Figura 5. Diagrama de clases del diseño HU\_2 Gestionar Problema de Investigación (Elaboración propia)

### Descripción del diagrama de clases de diseño de Gestionar Problema de Investigación

Para gestionar un problema de investigación se realiza una petición desde la página principal a la página servidora donde se encuentra la clase controladora **CC\_ProbController**, que construye la página cliente **CP\_Problema**, la cual se compone por el formulario **Form\_CreateProblema** que permite el ingreso de los datos requeridos para la creación de los problemas de investigación y que será ejecutado por la página servidora.

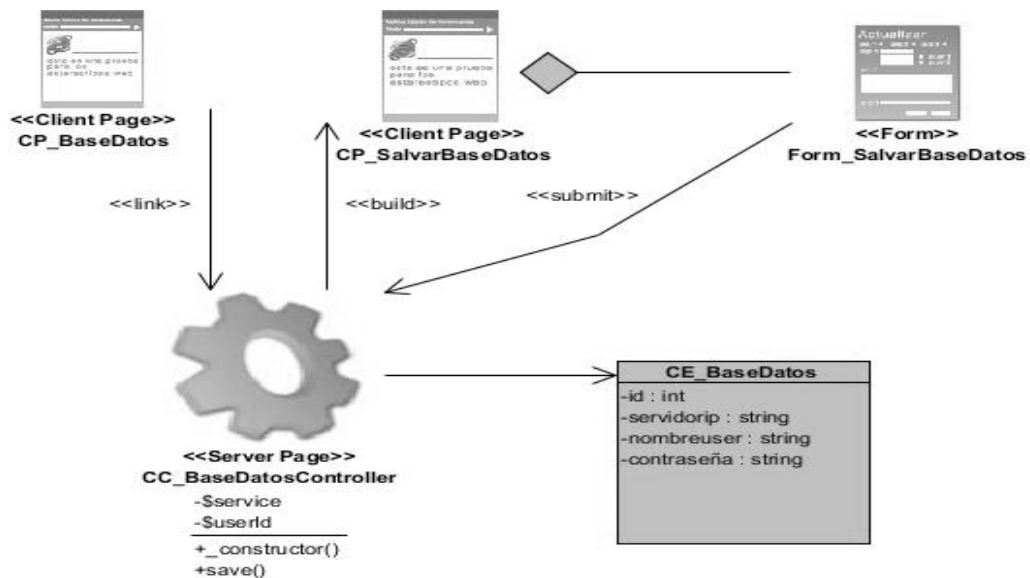


Figura 6. Diagrama de clases del diseño HU\_5 Salvar Base de Datos (Elaboración propia)

### Descripción del diagrama de clases de diseño Salvar Base de Datos

Para salvar la base de datos se realiza una petición desde la página principal a la página servidora donde se encuentra la clase controladora `CC_BaseDatosController`, que construye la página cliente `CP_SalvarBaseDatos`, la cual se compone por el formulario `Form_SalvarBaseDatos` que permite el ingreso de los datos requeridos para la salva de la base de datos y que será ejecutado por la página servidora.

## 2.8 Diagrama de secuencia

Los diagramas de secuencia (DS) en el UML se utilizan principalmente para modelar las interacciones entre los actores y los objetos en un sistema, así como las interacciones entre los objetos (Sommerville, 2011). Para la presente investigación se generaron un total de cinco (5) DS relacionados con los requisitos funcionales de prioridad alta, definidos en el subepígrafe 2.3.1. A continuación se muestran dos (2) relacionado con el RF 2 y el RF 5, el resto puede ser consultado en el Anexo 3.

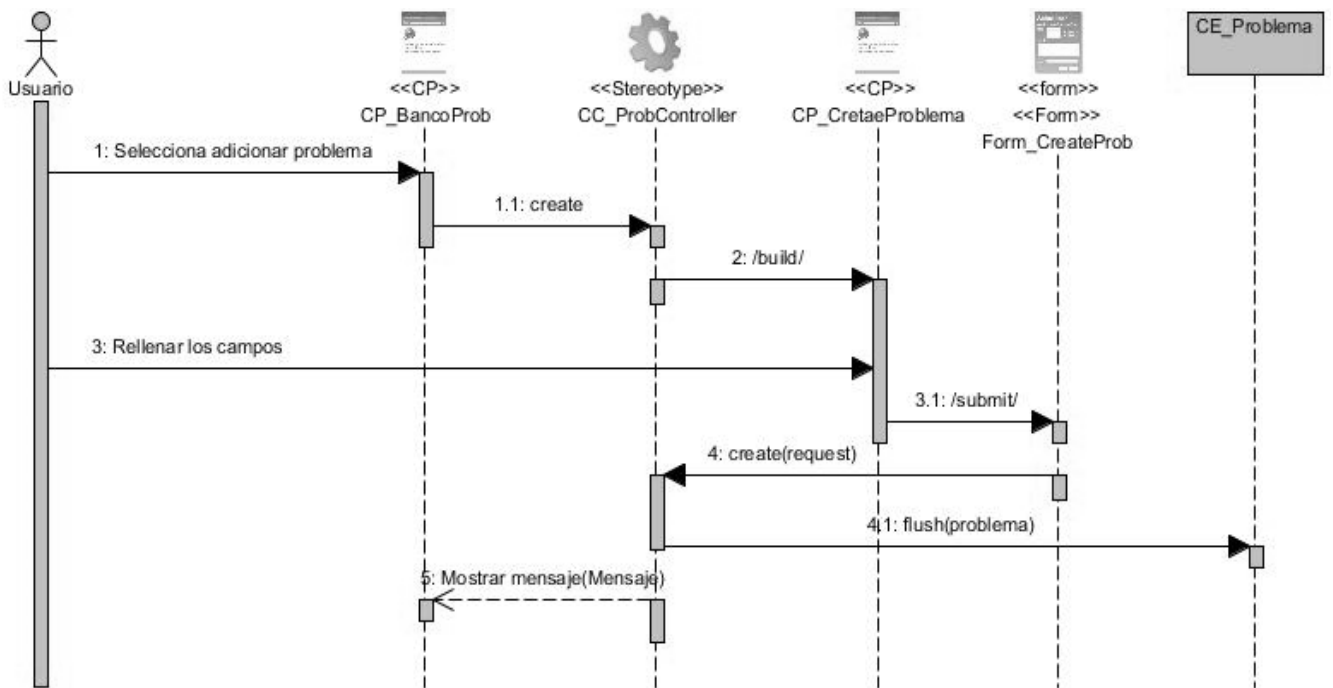


Figura 7. Diagrama de secuencia HU Gestionar Problema de Investigación (Elaboración propia)

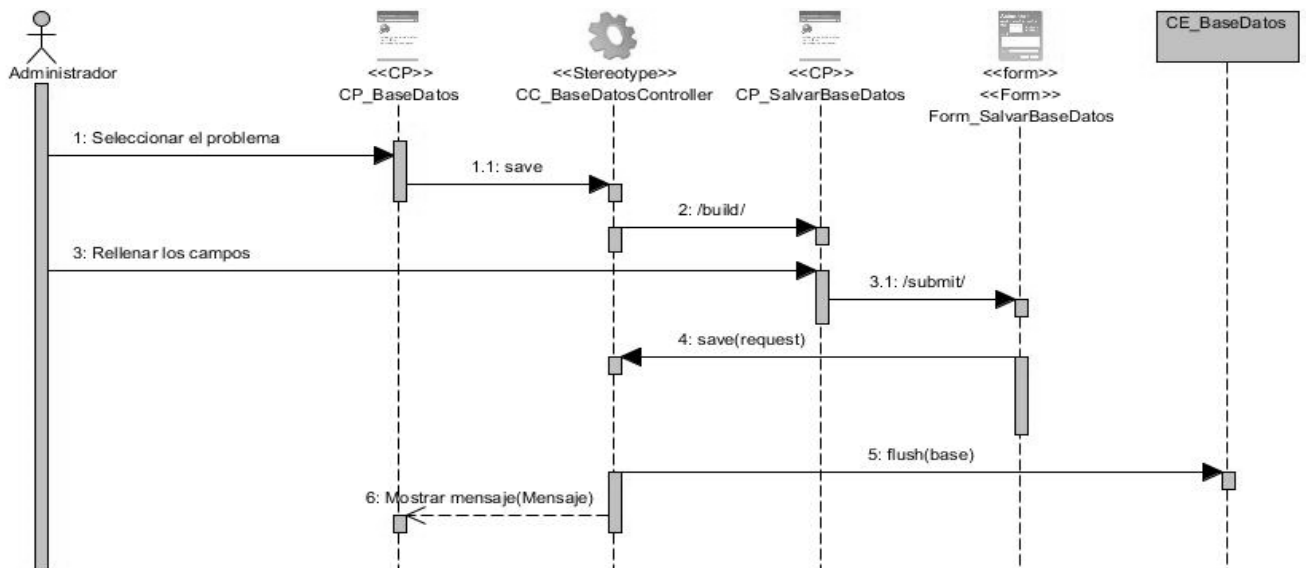


Figura 8. Diagrama de secuencia HU Salvar Base de Datos (Elaboración propia)

## 2.9 Modelo de datos

Un modelo de base de datos muestra la estructura lógica de la base, incluidas las relaciones y limitaciones que determinan cómo se almacenan los datos, la relación que existe entre sí, los procesos que los transforman y cómo se accede a ellos. Se basa en la identificación de los objetos primarios que va a procesar el sistema, la composición y atributos de los mismos. En algunos casos, esta base de datos es independiente del sistema software (Sommerville, 2011). El modelo de datos generado para la propuesta de solución de la presente investigación, como se observa en la Figura 5, está estructurado en dos (2) tablas, estas se explican a continuación:

**Usuario:** Registra la información referente a los usuarios del sistema, de los mismos almacena los atributos: identificador, nombre, apellidos, usuario, contraseña, semilla para encriptación de la contraseña, el rol que desempeña.

**Problema de Investigación:** Registra la información de los problemas de investigación, de los mismos se almacena los atributos: identificador, nombre, línea científica, clasificación, descripción.

**Línea Científica:** Registra la información referente a los distintos temas que puede abordar un problema de investigación, de ella se almacenan el siguiente atributo: nombre.

**Rol:** Registra la información referente a los distintos roles que puede adoptar un usuario, de este se almacenan el siguiente atributo: nombre.

**Trazas:** Registra la información referente a las trazas que dejan los usuarios al entrar al sitio, de ella se almacenan los siguientes atributos: IP, fecha, usuario.



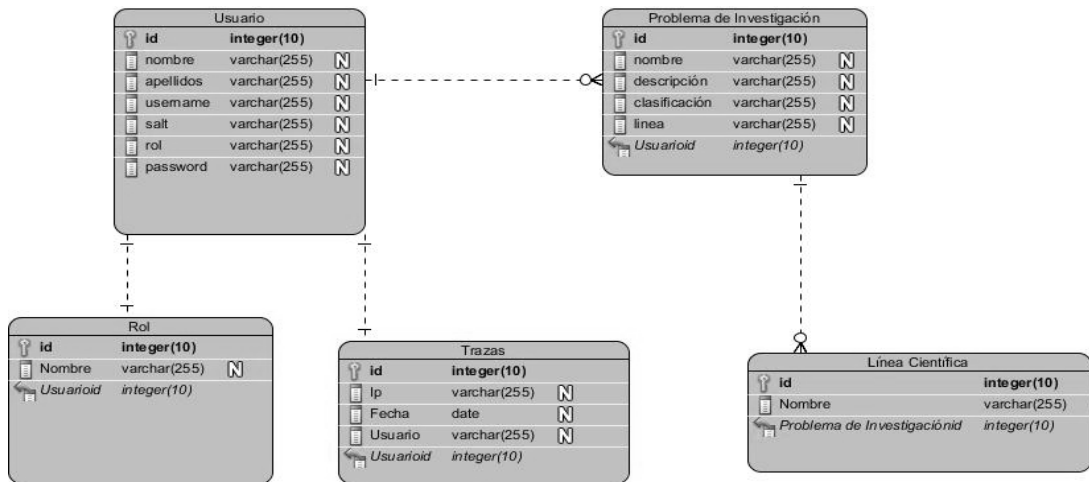


Figura 9. Modelo de datos de la propuesta de solución. (Elaboración propia)

## 2.10 Modelo de despliegue

Según Sarmiento (2016), un modelo de despliegue es un diagrama estructurado que muestra la arquitectura del sistema desde el punto de vista de distribución de los artefactos del software en los destinos de despliegue; se definen los artefactos como representaciones de elementos concretos en el mundo físico que son el resultado de un proceso de desarrollo. La Figura 6 muestra el despliegue propuesto para el sistema:

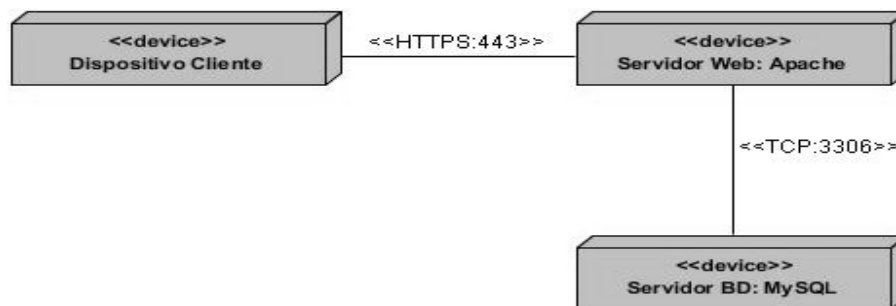


Figura 10. Modelo de despliegue (Elaboración propia)

En este diagrama se define el nodo Dispositivo Cliente desde donde se originan las peticiones mediante el protocolo *HTTPS* a la aplicación web alojada en el servidor de aplicaciones web *Apache*. Este servidor debe utilizar un sistema operativo cuyas propiedades mínimas sean un procesador de 2.10 GHz y 1GB de

memoria *RAM*. El sistema mantendrá una comunicación con el servidor de Base de Datos a través del protocolo *TCP* por el puerto 3306, este utilizará el SGBD *MySQL*.

## **2.11 Conclusiones del capítulo**

En este capítulo se realizó el análisis, el diseño y la arquitectura en general del sistema para la gestión del banco de problemas de investigación de la Facultad 1. Para cumplir dicha tarea se representaron y describieron los artefactos definidos por la metodología de manera que se entendiera el flujo de trabajo presente. Se representó el entorno en el que se ubica el problema a través del modelo conceptual, que constituye un punto de partida para el diseño del sistema. Los requisitos funcionales y no funcionales obtenidos sirvieron de guía para desarrollar las distintas funcionalidades y de este modo satisfacer las necesidades detectadas. Los artefactos generados según la metodología de desarrollo utilizada y los patrones de arquitectura y diseño descritos, constituyeron una guía fundamental para la construcción de la propuesta de solución.

## Capítulo 3: Implementación y validación del sistema para la gestión del banco de problemas científicos de la Facultad 1.

Con el objetivo de asegurar que el módulo propuesto funciona de acuerdo a los requisitos definidos y materializar en forma de componentes la fase de análisis y diseño, en el presente capítulo se describe la etapa de implementación y codificación del módulo. Se documentan los resultados obtenidos al aplicar las estrategias de pruebas, utilizadas para comprobar la calidad del software durante la etapa de validación.

### 3.1 Diagrama de Componentes

El diagrama de componentes proporciona una visión física de la construcción del sistema. Muestra la organización de los componentes software, sus interfaces y las dependencias entre ellos (Ramos, 2016). Basándose en la arquitectura de software que propone el marco de trabajo Symfony, seleccionado en la fase de análisis y diseño, la Figura 9 muestra el diagrama de componentes del sistema para la gestión del banco de problemas científicos de la Facultad 1, a continuación se describen los elementos que componen el diagrama:

**routing.yml:** Contiene el mapa de rutas *URL* que se enlazan a las acciones de los controladores del sistema.

**config.yml:** Contiene las configuraciones generales del sistema.

**Modelo:**

**ProblemaInvestigacion.php:** Clase entidad que almacena los datos de los problemas de investigación.

**LineaCientifica.php:** Clase entidad que almacena los datos de las líneas científicas a las que están asociados los problemas de investigación.

**Controlador:**

**ProblemaInvestigacionController.php:** Clase controladora que contiene las acciones relacionadas con los problemas de investigación. Realiza acciones generales de gestión (insertar, eliminar, mostrar, editar y buscar), aprobar.

**LineaCientificaController.php:** Clase controladora que contiene las acciones relacionadas con las líneas científicas. Realiza acciones generales de gestión (insertar, eliminar, mostrar, editar y buscar).

**UsuarioController.php:** Clase controladora encargada de gestionar los usuarios.

**Vista:**

**Formulario:**

**ProblemaInvestigacionType.php:** Clase responsable de construir los formularios relacionados con los problemas de investigación.

**LineaCientificaType.php:** Clase responsable de construir los formularios relacionados con las líneas científicas asociadas a los problemas de investigación.

**Problema de Investigación:**

**index.html.twig:** Muestra un listado de los problemas de investigación registrados en el sistema.

**new\_prob.html.twig:** Permite insertar un problema de investigación en el sistema a través de un formulario.

**edit\_prob.html.twig:** Permite editar un problema de investigación a través de un formulario.

**Línea Científica:**

**index.html.twig:** Muestra un listado con las líneas científicas registradas en el sistema.

**new\_linea.html.twig:** Permite registrar una línea científica en el sistema.

**edit\_linea.html.twig:** Permite editar una línea científica en el sistema.

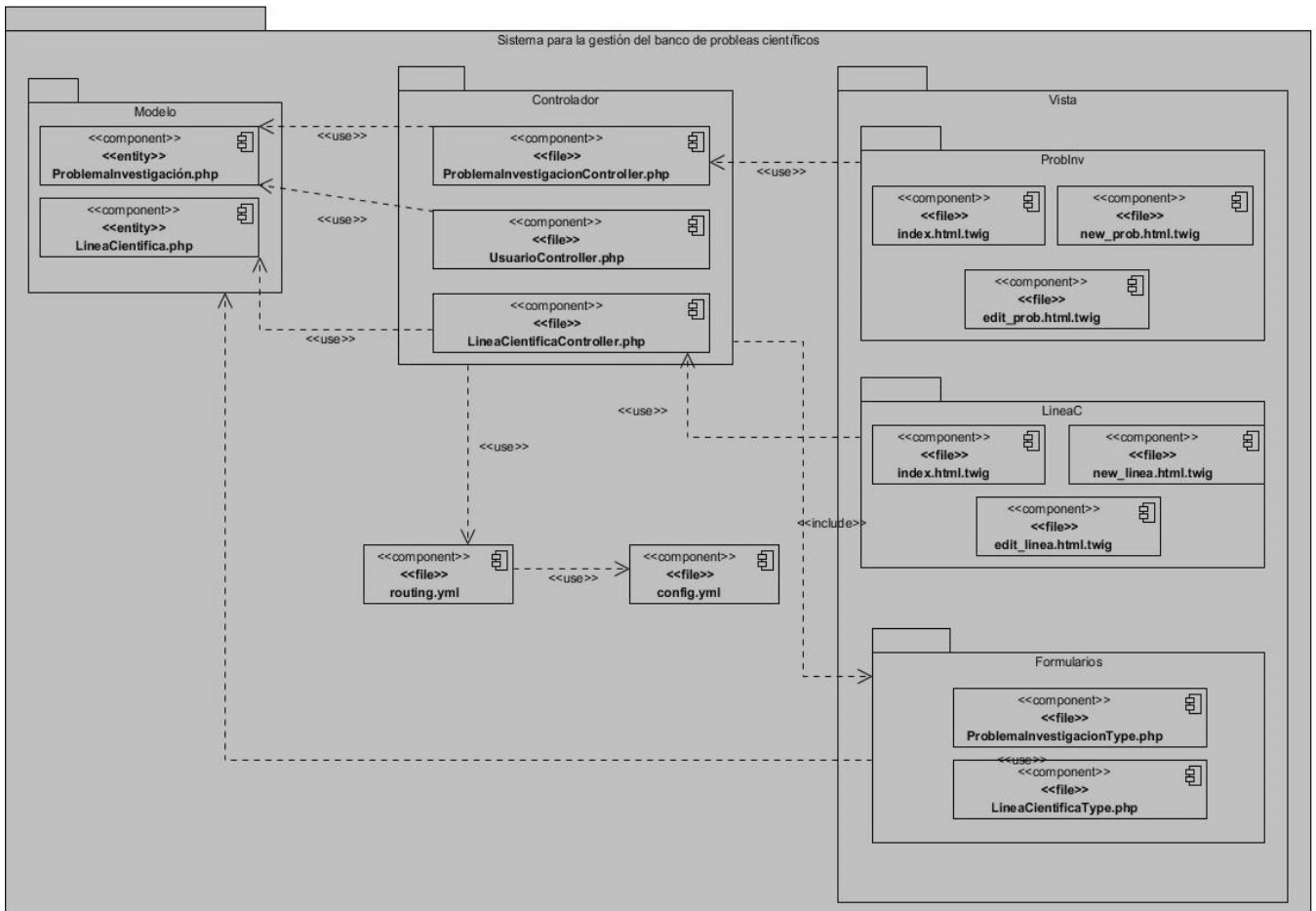


Figura 11. Diagrama de componentes (Elaboración propia)

### 3.2 Estándares de Codificación

Los estándares de codificación son un conjunto de convenciones para lograr uniformidad en el código de fuente de un software. Permiten una mayor comprensión, modificación, calidad y mantenibilidad del código generado independientemente de su autor. A continuación, se definen los estándares de codificación utilizados durante la implementación del módulo, basándose en los que propone el lenguaje *PHP* para el marco de trabajo *Symfony* (PHP-Fig, 2018).

Tabla 5. Estándares de codificación (Elaboración propia)

Tipo de Estándar	Descripción
Organización del código	<ol style="list-style-type: none"> <li>1. La apertura de llaves para clases y funciones deben estar en la próxima línea después de su declaración.</li> <li>2. Las aperturas de llaves en las estructuras de control se colocan en la misma línea.</li> <li>3. El cierre de llaves para clases, funciones y estructuras de control deben estar en la próxima línea después del bloque de código.</li> <li>4. La indentación se realiza con tabulación y no con 4 espacios en blanco.</li> </ol>
Líneas	<ol style="list-style-type: none"> <li>5. El tamaño máximo de una línea no excederá de ochenta (80) caracteres.</li> <li>6. Se deberá agregar una línea en blanco delante de la sentencia <i>return</i>.</li> <li>7. Se puede agregar líneas en blanco para mejorar la legibilidad e indicar bloques de códigos relacionados.</li> <li>8. Debe haber una declaración por línea.</li> </ol>
Codificación	<ol style="list-style-type: none"> <li>9. Los archivos <i>PHP</i> deben utilizar solo las etiquetas de apertura <code>&lt;?php</code> omitiendo las etiquetas de cierre <code>?&gt;</code></li> <li>10. Utilizar la codificación <i>UTF-8</i>.</li> </ol>
Espacios en blanco en expresiones y sentencias	<ol style="list-style-type: none"> <li>11. Se debe utilizar un espacio en blanco después de la palabra clave en las estructuras de control.</li> <li>12. No deben existir espacios en la apertura y cierre de paréntesis.</li> <li>13. Debe usarse un espacio en blanco entre los operadores lógicos, aritméticos, de comparación y asignación</li> </ol>
Convenciones de Nombres	<ol style="list-style-type: none"> <li>14. Se debe utilizar el estilo de escritura <i>camelCase</i> para la declaración de variables y no guiones bajos para separar palabras.</li> <li>15. Utilizar letras mayúsculas sostenidas para declarar contantes y separar las palabras con guiones bajos.</li> <li>16. Se debe usar el estilo de escritura <i>StudlyCaps</i> para los nombres de clases y funciones.</li> </ol>

### 3.3 Validación de la propuesta de solución

Según la ISO/IEC/IEEE International Standard (2017), la fase de validación es el proceso de evaluar la calidad del software, con el objetivo de detectar posibles errores y corregirlos enfocándose en la lógica interna y los requerimientos especificados. A continuación, se documentan los resultados de las pruebas realizadas al Sistema para la gestión del banco de problemas científicos de la Facultad 1, con el objetivo de evaluar su calidad y correcto funcionamiento.

### 3.3.1 Pruebas Funcionales

Las pruebas funcionales tienen como objetivo validar que las funcionalidades implementadas cumplan con las especificaciones de los requisitos definidos. Para esta prueba, el autor selecciona la técnica de Caja Negra, que según Pressman (2010) permite derivar un conjunto de condiciones de entrada, que revisarán por completo todos los requisitos funcionales de una aplicación con el objetivo encontrar errores de funciones incorrectas o ausentes, errores de interfaz, de comportamiento o rendimiento. Para aplicar las pruebas funcionales al sistema se diseñaron cinco (5) Casos de Prueba (CP) que corresponden a los requisitos funcionales de prioridad alta. A continuación, se muestra un (1) CP correspondiente al RF6, el resto puede ser consultado en el Anexo 4.

Tabla 6. Descripción de las variables para el Caso de Prueba 1 (Elaboración propia)

Nº	Nombre del campo	Clasificación	Valor Nulo	Descripción
1	Título del problema	Campo de texto	No	Permite todos los caracteres. Longitud máxima 50.
2	Descripción	Campo de texto	No	Permite todos los caracteres. Longitud máxima 200.
3	Clasificación	Campo de selección	No	Puede tomar los siguientes valores maestría, doctorado, tesis.
4	Línea Científica	Campo de selección	No	Puede tomar los siguientes valores inteligencia artificial, redes de computadoras, computación gráfica, software libre, computación científica, informática aplicada en la sociedad.

Tabla 7. Caso de Prueba del RF2\_Gestionar un problema de investigación (Elaboración propia)

Caso de Prueba 1: SC RF2_Gestionar un problema de investigación							
Condiciones de ejecución: El usuario debe estar autenticado en el sistema							
Escenario	Descripción	1	2	3	4	Respuesta del Sistema	Flujo Central

EC 1.1 Crear un problema de investigación de forma correcta	El sistema inserta un nuevo problema de forma correcta.	V	V	V	V	Muestra un mensaje de confirmación "Insertado satisfactoriamente el problema de investigación".	En el menú lateral del sistema:  1. El usuario accede a la opción Banco de Problemas y el sistema muestra una lista de los problemas de investigación registrados.
		Sistema para gestión del banco de problemas científicos	Un sistema que gestiona las diferentes problemáticas originadas en la Facultad 1	tesis	Redes de computadoras		
EC 1.2 Crear un problema de investigación de forma incorrecta	El sistema no inserta un nuevo problema de forma incorrecta.	V	V	I	V	Muestra un mensaje de error "Indique la clasificación", sale una notificación en rojo.	2. El usuario selecciona la opción Adicionar. 3. El usuario introduce los datos en el formulario que se muestra. 4. Se presiona el botón adicionar.
		Sistema para gestión del banco de problemas científicos	Un sistema que gestiona las diferentes problemáticas originadas en la Facultad 1		Redes de computadoras		

### Resultado de las pruebas funcionales

Se obtuvo en una primera iteración un total de ocho (8) no conformidades, divididas en cinco (5) de ortografía, las cuales fueron corregidas al momento; dos (2) de funcionalidad relacionadas con la ausencia de la notificación al usuario cuando los datos de los campos son incorrectos, para resolverlo se incluyen los mensajes antes mencionados al código de fuente. Una no conformidad de excepción que se produce en la adición de un problema; al introducir incorrectamente los datos en los campos con caracteres extraños o letras, se muestra un error fatal que interrumpe el flujo normal del sistema, fue resuelta al validar que en los campos Título y Descripción el tipo de dato sea de texto. Las deficiencias identificadas fueron resueltas en la primera iteración. Para una segunda iteración no se identificaron



nuevas no conformidades obteniendo resultados satisfactorios y un sistema funcional que cumple los requisitos especificados. Los resultados son expuestos en la Figura 10.

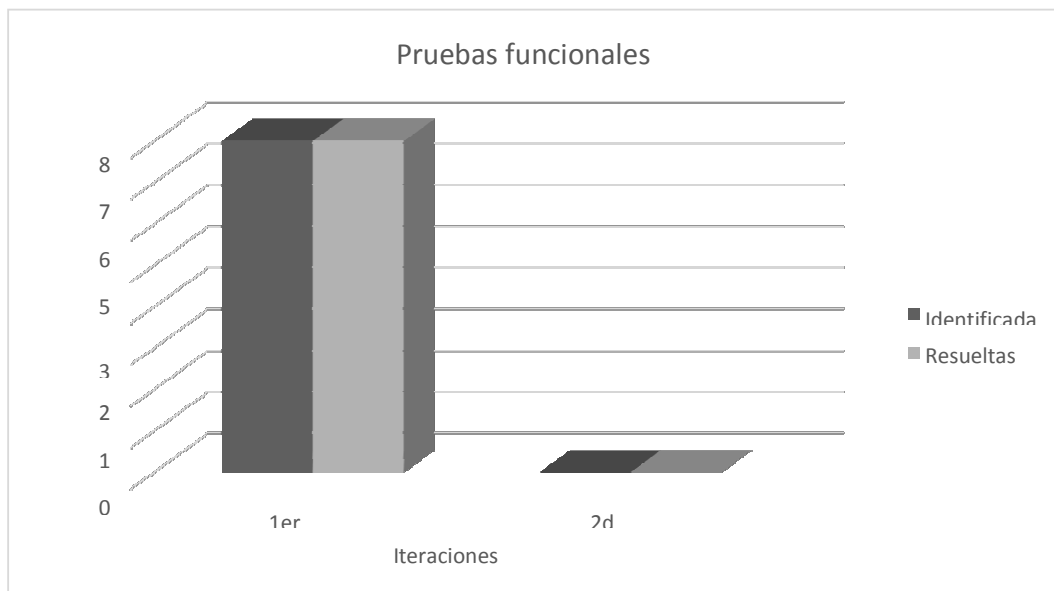


Figura 12. Resultado de las pruebas funcionales (Elaboración propia)

### 3.3.2 Pruebas de Seguridad

Las pruebas de seguridad se diseñan para identificar las vulnerabilidades que presenta el sistema. Cada uno de estos dominios pueden atacarse, ocasionando pérdidas para la entidad y es tarea del examinador de seguridad descubrir las debilidades que puedan explotar quienes tengan intención de hacerlo (Pressman, 2010).

Para realizar las pruebas de seguridad se utilizó la herramienta *Acunetix Web Vulnerability Scanner*. Mediante estas pruebas se detectaron cincuenta y ocho (58) alertas de ellas cuarenta y nueve (49) son de nivel medio y nueve (9) de carácter informacional. De las vulnerabilidades de nivel medio cuarenta y ocho (48) están relacionadas con los mensajes de error producidos en el modo de desarrollo del marco de trabajo Symfony y se detectó un formulario que no está protegido contra la falsificación de solicitudes entre sitios (CSFR por sus siglas en inglés). Las alertas de carácter informativo se dividen en ocho (8) enlaces rotos y una relacionada con el fichero

*parameters.yml* que contiene los atributos para la conexión con el sistema gestor de base de datos MySQL. La alerta del formulario se soluciona manualmente como recomienda la herramienta *Acunetix Web Vulnerability*, denominándose un falso positivo y el resto se resuelve cuando se pasa el sistema al modo producción. Luego de ser solucionadas las alertas de vulnerabilidad encontradas se obtiene un sistema seguro que cumple con los requisitos de seguridad avalados por la Dirección de Calidad de la UCI.

### 3.3.3 Pruebas de Integración

Las pruebas de integración se realizan con el objetivo de asegurar que todos los componentes del sistema se integren entre sí de forma correcta. Estas pruebas implican construir la aplicación desde sus componentes independientes y probar el sistema resultante en busca de errores que puedan surgir con el proceso de integración (Sommerville, 2011). Con el objetivo de validar la compatibilidad y el funcionamiento de las distintas funcionalidades que componen al Sistema para la gestión del banco de problemas científicos de la Facultad 1 se toman acciones relacionadas con:

- Validación de conexión a la base de datos (Ver Tabla 8).

Tabla 8. Caso de Prueba 2 Salvar base de datos (Elaboración propia)

<b>Caso de Prueba 2: RF5_Salvar base de datos.</b>			
<b>Condiciones de ejecución:</b> El usuario (administrador) debe estar autenticado en el sistema y debe haber llenado correctamente el formulario para salvar la base de datos.			
<b>Escenario</b>	<b>Descripción</b>	<b>Respuesta del sistema</b>	<b>Flujo Central</b>
EC 2.1. Salvar la base de datos de forma correcta	El sistema salva la base de datos de forma correcta, luego de haber validado los campos del formulario para salvar la misma y comprobar la conexión del sistema a ella.	Muestra un mensaje de confirmación: "La base de datos fue salvada correctamente".	1. El administrador accede a la opción Salvar base de datos. 2. El sistema muestra un formulario para salvar la base de datos. 3. El usuario introduce valores válidos y presiona el botón Salvar.

En la aplicación de esta prueba no se identificaron no conformidades, lo que valida que existe una correcta integración de los componentes internos del sistema.

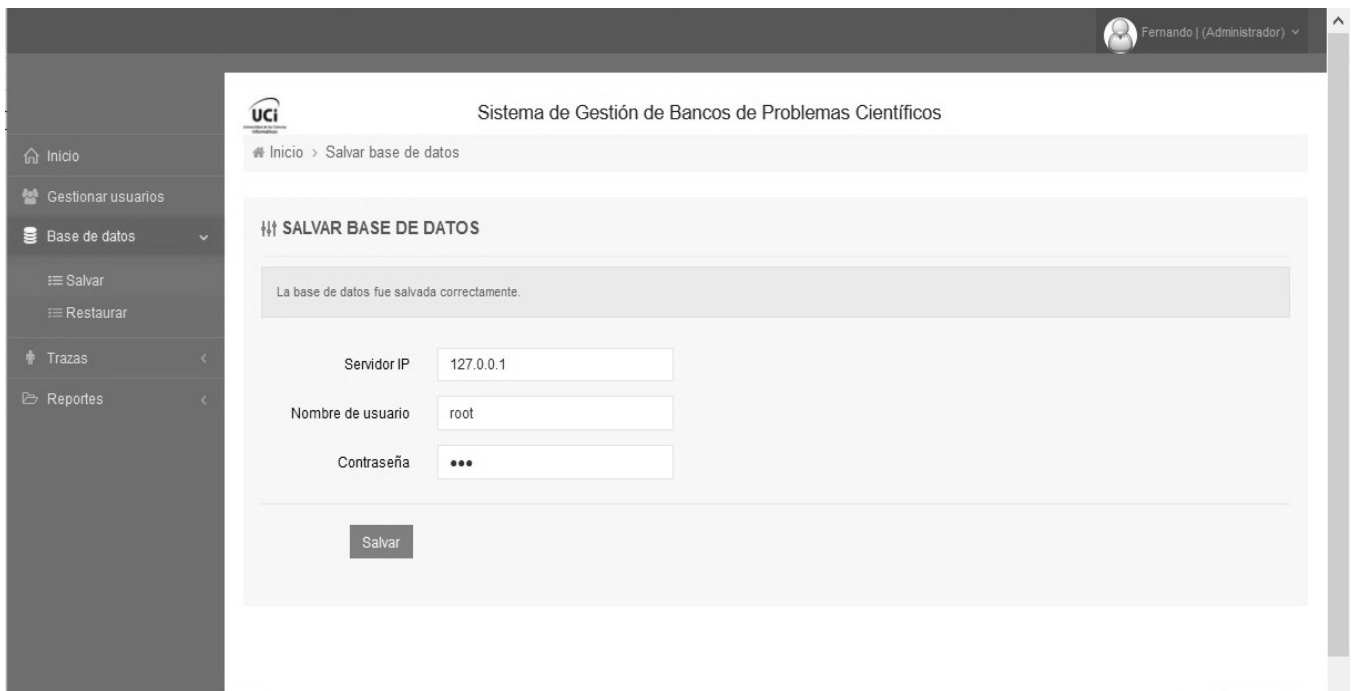


Figura 13. Resultado de las pruebas de integración (Elaboración propia)

### 3.3.4 Pruebas de Rendimiento

Las pruebas de rendimiento de software se centran en determinar la velocidad con la que el sistema bajo pruebas realiza una tarea en las condiciones particulares del escenario de pruebas (Globe, 2018).

#### Carga

Mediante la ejecución de las pruebas de Carga es posible identificar la capacidad de recuperación de un sistema cuando es sometido a cargas variables tanto de usuarios como de procesos. Al realizar las pruebas de carga se puede determinar el tiempo de respuesta de todas las transacciones críticas del sistema y encontrar cuellos de botella de la misma (V&v Quality S.A, 2016).

## **Estrés**

Mediante las pruebas de estrés es posible identificar la capacidad de respuesta de un sistema bajo condiciones de carga extrema, representadas por una alta concurrencia de Usuarios y/o procesos, una vez realizadas las pruebas de estrés se podrá conocer el punto de quiebre del aplicativo en términos de capacidad de respuesta, con lo cual será posible establecer acciones de optimización en diferentes niveles para asegurar una mejor capacidad de concurrencia de usuarios y/o procesos que se verá reflejada en una óptima operación de negocio (V&v Quality S.A, 2016).

## **Resultados de las pruebas de rendimiento**

Para las pruebas de rendimiento se utiliza el software Apache JMeter v2.8.4. Para ello se definen las propiedades de las PC utilizadas tanto la cliente como la utilizada como servidor.

### **Hardware de prueba (PC cliente):**

- Tipo de procesador: Intel(R) Core(TM) 2 Duo CPU P8400 @2.26GHz 2.27GHz.
- RAM: 4 GB.
- Tipo de Red: Ethernet 10/100Mbps.

### **Hardware de prueba (PC servidor):**

- Tipo de procesador: Intel(R) Core(TM) 2 Duo CPU P8400 @2.26GHz 2.27GHz.
- RAM: 4 GB.
- Tipo de Red: Ethernet 10/100Mbps.

### **Software instalado en ambas PC:**

- Tipo de servidor web: Apache 2.4.
- Plataforma: SO Linux (PC servidor) y SO Windows (PC cliente).
- Servidor de BD: MySQL 5.0

## **Resultado de las pruebas de rendimiento**

- **Etiqueta:** El nombre de la muestra (conjunto de muestras).
- **# Muestras:** El número de muestras para cada URL (peticiones).

- **Mín:** El mínimo tiempo (milisegundos) transcurrido para las muestras de la URL dada.
- **Máx:** El máximo tiempo (milisegundos) transcurrido para las muestras de la URL dada.
- **% Error:** Porcentaje de las peticiones con errores.
- **Rendimiento:** Rendimiento medido en base a peticiones respondidas por segundo/minuto/hora.

Tabla 9: Informe agregado

#Muestras	Min	Max	% Error	Rendimiento (peticiones/segundos)
50	3535 ms	10753 ms	0.00	3.1/sec

Según Nielsen, 1993 se considera un buen tiempo de respuesta 1.0 a 10.0 segundos para una aplicación dinámica. Este es aproximadamente el límite para que el flujo de pensamiento del usuario permanezca ininterrumpido, aunque el usuario notará la demora. Luego de analizados los datos de la tabla 8 se puede afirmar que el Sistema para la gestión del banco de problemas científicos funciona correctamente y en un tiempo aceptable, pues el sistema no devuelve ningún error al realizarse estas peticiones y responde en un buen tiempo cada una de estas.

### 3.3.5 Validación de la hipótesis científica de la investigación

Con el objetivo de evaluar la fiabilidad de la hipótesis científica de la investigación se aplica el Criterio de Expertos mediante el método Delphi. Este método consiste en una técnica de obtención de información, basada en la consulta de expertos en un área de estudio, con el fin de obtener la opinión de consenso más fiable proveniente del conocimiento y la experiencia de los participantes del grupo (Reguant y otros, 2016).

De la hipótesis científica planteada en la presente investigación, se define como variable independiente: el sistema para la gestión del banco de problemas científicos de la Facultad 1, consiste en un sistema que permita gestionar las distintas problemáticas originadas en la facultad. Como variable dependiente: mejorar el proceso de gestión de los problemas de investigación, refiriéndose a la disminución del tiempo, esfuerzo y facilidad en las tareas de gestión del sistema.

Para aplicación del Método Delphi se emplearán los siguientes pasos:

- Identificación y selección de posibles expertos.
- Aplicación de encuestas a expertos.
- Valoración de la información obtenida.

#### Identificación y selección de posibles expertos

Se identifican cinco (5) posibles expertos en materias relacionadas con la gestión de problemas científicos. Con el objetivo de valorar el nivel de experiencia que poseen, se aplica un cuestionario de autovaloración (ver Anexo 5) de los niveles de información y argumentación que tienen sobre el tema en cuestión. En la Tabla 10 se resumen los resultados obtenidos.

Tabla 10. Resultado de niveles de conocimiento de posibles expertos (Elaboración propia)

Expertos	Nivel de conocimiento o información del tema										Kc
	1	2	3	4	5	6	7	8	9	10	
Miguel Ángel Hernández de la Rosa								X			0.8
Allan Pierra Conde							X				0.7
Yaniel Lázaro Aragón Barreda									X		0.9
Delly Lien González Hernández							X				0.7
Yaimí Trujillo Casañola							X				0.7

Para calcular el Coeficiente de Conocimiento o Información (Kc) del experto se utiliza la ecuación 3.1:

$$Kc = n(0.1) \quad (3.1)$$

Donde  $n$  es el nivel de conocimiento o información seleccionado por el experto. Para determinar, además, el coeficiente de argumentación o fundamentación de cada experto es necesario utilizar como factores, los que aparecen en la Tabla 11.

Tabla 11. Patrón para el cálculo de Coeficiente de Argumentación o Fundamentación  
(Naranjo, 2018)

Fuentes de Argumentación	Alto	Medio	Bajo
Análisis teóricos realizados por usted	0.3	0.2	0.1
Experiencia obtenida	0.5	0.4	0.2
Trabajos de autores nacionales	0.05	0.05	0.05
Trabajos de autores extranjeros	0.05	0.05	0.05
Su conocimiento del estado del problema en el extranjero	0.05	0.05	0.05
Su Intuición	0.05	0.05	0.05

El Coeficiente de Argumentación ( $K_a$ ) de cada experto se calcula con la ecuación 3.2:

$$K_a = \frac{\sum n_i}{n} \quad (3.2)$$

Donde  $n_i$  es el valor obtenido en la fuente de argumentación  $i$  (de 1 hasta  $n$ ) y  $n$  es la cantidad de fuentes de argumentación. Al aplicar la ecuación se obtienen los siguientes resultados:

- Miguel Ángel Hernández de la Rosa.  $K_a = 0.9$
- Allan Pierra Conde.  $K_a = 0.8$
- Yaniel Lázaro Aragón Barreda.  $K_a = 0.8$
- Delly Lien González Hernández.  $K_a = 0.9$
- Yaimí Trujillo Casañola.  $K_a = 0.8$

Con los Coeficientes de Conocimiento ( $K_c$ ) y Argumentación ( $K_a$ ) se puede obtener finalmente el Coeficiente de Competencia ( $K$ ) para determinar que expertos se toman en consideración para trabajar en la investigación. Este es calculado con la ecuación 3.3:

$$K = 0.5 (K_c + K_a) \quad (3.3)$$

El resultado obtenido se muestra en la Tabla 12 y es valorado de la siguiente manera:

- El coeficiente es alto si  $0.8 < K < 1$

- El coeficiente es medio si  $0.5 < K < 0.8$
- El coeficiente es bajo si  $K < 0.5$

Tabla 12. Resultado de niveles de competencia de posibles expertos (Elaboración propia)

Expertos	K	Valoración
Miguel Ángel Hernández de la Rosa	0.85	Alto
Allan Pierra Conde	0.7	Medio
Yaniel Lázaro Aragón Barreda	0.85	Alto
Delly Lien González Hernández	0.8	Alto
Yaimí Trujillo Casañola	0.75	Medio

Fueron seleccionados tres expertos al ser su nivel de competencia alto, finalmente el panel de expertos quedó conformado de la siguiente manera (Ver Tabla 13):

Tabla 13. Expertos seleccionados para la validación de la hipótesis científica (Elaboración propia)

Nombre y Apellidos	Entidad	Años de Experiencia
Delly Lien González Hernández	Dirección de Relaciones Internacionales (UCI)	13
Yaniel Lázaro Aragón Barreda	Vicedecanato de Investigación y Postgrado (Facultad 1)	6
Miguel Ángel Hernández de la Rosa	Vicedecanato de Investigación y Postgrado (Facultad 1)	5

Aplicación de encuestas a expertos:

Seleccionado el panel de expertos que trabajará en la validación de la hipótesis científica de la investigación, se somete a su consideración una encuesta (Ver Anexo 6), para evaluar el Sistema para la gestión del banco de problemas científicos de la Facultad 1. La Tabla 14 resume el resultado de los juicios emitidos por los expertos, de acuerdo a las siguientes sentencias:



1. Facilidad en la gestión de los problemas de investigación.
2. Facilidad en la asignación de temas de tesis.
3. Rapidez en la respuesta de peticiones.
4. Presentación de una interfaz agradable e intuitiva para el usuario.

Para el procesamiento y análisis de la información obtenida se clasificaron las respuestas en las siguientes categorías evaluativas: muy adecuado (MA), adecuado (A), poco adecuado (PA), inadecuado (I).

Tabla 14. Resultado de las encuestas realizadas a los expertos (Elaboración propia)

Sentencias	Categorías Evaluativas				Total
	MA	A	PA	I	
1	3	0	0	0	3
2	2	1	0	0	3
3	1	2	0	0	3
4	2	1	0	0	3

Expuestos los criterios de los expertos en cada categoría evaluativa para las sentencias de la encuesta, se aplican los siguientes pasos hasta llegar a la conclusión de valoración de cada sentencia:

- Obtención de la tabla de frecuencia acumulada (Ver Tabla 15).
- Obtención de la tabla de frecuencia relativa acumulativa (Ver Tabla 16).
- Asignación del valor de la imagen que corresponde a cada frecuencia relativa acumulativa, por la inversa de la curva normal (Tabla Z de la distribución normal) y obtención de puntos mediante el cálculo N-P para el promedio relativo (Ver Tabla 17).

Tabla 15. Frecuencia acumulada de los datos primarios obtenidos (Elaboración propia)

Sentencias	Categorías Evaluativas			
	MA	A	PA	I
1	3	3	3	3
2	2	3	3	3
3	1	3	3	3
4	2	3	3	3

Tabla 16. Frecuencia relativa acumulativa de los datos primarios obtenidos  
(Elaboración propia)

Sentencias	Categorías Evaluativas	
	MA	A
1	1	1
2	0.6	1
3	0.3	1
4	0.6	1

Tabla 17. Imagen de la frecuencia relativa acumulativa de los datos primarios  
obtenidos (Elaboración propia)

Sentencias	Categorías Evaluativas		Suma	Promedio	Promedio relativo
	MA	A			
1	3.49	3.49	6.98	3.49	-1.29
2	0.27	3.49	3.76	1.88	0.31
3	- 0.43	3.49	3.06	1.53	0.66
4	0.27	3.49	3.76	1.88	0.31
Puntos de Corte	3.6	3.49	17.56		

Al analizar los promedios relativos (N-P) de las sentencias, se puede observar que ninguno excede el límite superior o punto de corte de la categoría evaluativa MA. Esto demuestra que los expertos con un nivel de concordancia del 100% clasifican con el más alto nivel de adecuación (Muy Adecuado) las sentencias evaluadas, lo que apoya la hipótesis científica planteada en la presente investigación. Todos los indicadores fueron evaluados satisfactoriamente lo que evidencia la calidad y el alto valor del Sistema para la gestión del banco de problemas científicos de la Facultad 1.

### 3.4 Conclusiones del capítulo

- En este capítulo la confección del diagrama de componentes permitió establecer una mayor comprensión de la arquitectura del sistema.

- La utilización de los estándares de codificación de código permitió garantizar que el código posea alta calidad, menos errores y fácil mantenimiento.
- La aplicación de las pruebas de carga y estrés, funcionalidad, seguridad, rendimiento, integración y la consulta a expertos permitieron identificar las principales deficiencias en el desarrollo del sistema, así como la estrategia para solucionar los errores detectados y obtener un producto con un alto valor, pertinencia y utilidad.

## Conclusiones

- Los principales conceptos asociados al dominio de la investigación, posibilitaron lograr un mayor dominio referente a los sistemas de gestión de la información.
- Los patrones de diseño ofrecieron un mayor soporte a la implementación de los requisitos previamente expresados por el cliente; garantizando la estructura base para la organización lógica del código fuente y la disminución del impacto ante futuras modificaciones en la aplicación.
- La utilización de las herramientas y tecnologías identificadas y la puesta en práctica de los estándares de codificación definidos, permitieron implementar el Sistema para la gestión del banco de problemas científicos de la Facultad 1.
- Las pruebas de software solucionaron los errores detectados en el sistema y demuestran que es una solución funcional, integrable y segura.
- El criterio de expertos, como técnica de validación, demostró con un alto nivel de concordancia que la propuesta desarrollada facilita la asignación de los temas de tesis a estudiantes y mejora la gestión de los problemas científicos en la Facultad 1.

## Recomendaciones

Para el desarrollo de futuras investigaciones relacionadas con la presente, se recomienda:

- Realizar un foro donde la comunidad científica pueda interactuar según sus temas de interés.
- Adicionar una funcionalidad donde el sistema sea capaz de proponer problemas de investigación de acuerdo a los intereses de los usuarios.

## Bibliografía

- Acunetix.** Manual de usuario. [En línea]. Consulta: 3 enero 2019. Disponible en: <https://www.acunetix.com/wp-content/uploads/2012/10/wvsmanual.pdf>
- Álvarez, Sara.** Conceptos básicos y definiciones sobre programación [En línea]. 2016 [Fecha de consulta: 14 noviembre 2018]. Disponible en: <https://desarrolloweb.com/articulos/2357.php>.
- Andalucía, Consejería de Economía y Conocimiento de la Junta de.** 2018. Centro Informático Científico de Andalucía. [En línea] 2018. <https://sica2.cica.es/repositan/index.htm>.
- ApacheJMeter.** [En línea]. Consulta: 26 noviembre 2016. Disponible en: <http://jmeter.apache.org>
- Artologik. 2009.** Help Desk. [En línea] 2009. <https://artologik.com/>.
- Arume. 2015.** [En línea] 2015. <http://www.arumeinformatica.es/dudas/css/>.
- Axarnet. 2018** [En línea] 26 de Abril de 2018. <https://www.axarnet.es/blog/lenguajes-del-lado-del-servidor/>.
- Bessarabov, Max. 2018.** JetRuby. [En línea] 2 de Febrero de 2019. The Top Web Development Frameworks in 2018 – JetRuby.com.
- Bustos, Gustavo. 2018.** Hostinger. [En línea] 31 de Noviembre de 2018. <https://www.hostinger.es/tutoriales/que-es-apache/>.
- Carbonell, Joaquim. 2016.** *Cómo implantar un sistema de gestión del conocimiento.* 2016.
- Cáedenas, Lain.** Aplicación de Patrones de Diseño para Garantizar Alta Flexibilidad en el Software. Universidad César Vallejo [En línea]. 2016. vol. 7, no. 1. pp. 77-82. [Fecha de consulta: 15 febrero 2019]. ISSN: 1819-4575. Disponible en: <http://revistas.ucv.edu.pe/index.php/RTD/article/view/696>
- Castellanos, Arturo Rodríguez. 2001.** *La Gestión del Conocimiento científico-técnico en la universidad.* Lejona : s.n., 2001.
- Coders, Value. 2017.** [En línea] 1 de Marzo de 2017. <https://www.valuecoders.com/blog/technology-and-apps/symfony-vs-laravel-php-framework-choose/>.
- ConceptoDefinición.DE. 2018.** [En línea] 2018. <https://conceptodefinicion.de/html/>.

**Craig, Larman.** UML y patrones: una introducción al análisis y diseño orientado a objetos y al proceso unificado. 2.<sup>a</sup> ed. Madrid: Pearson Educación, 2003. pp. 590. ISBN: 84-205-3438-2.

**Culturación.** 2011. [En línea] 25 de Octubre de 2011. <http://culturacion.com/que-es-mysql/>.

**Davenport, T. H.** 1998. *Salvando el alma: centrada en el hombre y la gestión de la información.* . 1998, Harvard Business Review.

**Educacionales., Centro de Estudios.** 1999. *Glosario. Taller de Diseño de Proyectos de Investigación-Desarrollo e Innovación Tecnológica. Instituto Superior Pedagógico Enrique José Varona.* s.l. : Facultad de Ciencias de la Educación, 1999.

**Espinoza, Angela Gabriela Torres.** 2012. SlideShare. [En línea] 5 de Junio de 2012. [www.Bancos-de-informacion.htm/](http://www.Bancos-de-informacion.htm/).

**Globe.** [En Línea]. Pruebas de rendimiento. Consulta: 3 mayo 2019. Disponible en: HYPERLINK "https://www.globetesting.com/pruebas-de-rendimiento/" <https://www.globetesting.com/pruebas-de-rendimiento/>.

**Gómez, Luis Carlos, y otros.** 2009. MinEducaión. [En línea] 31 de Marzo de 2009. <http://www.mineduccion.gov.co/1621/article-87646.html>

**Hamon, Hugo.** Applying Design Patterns to Symfony [En línea]. Estambul, Turquía: SensioLabs, 2014 [Fecha de consulta: 12 enero 2019]. Disponible en: <https://speakerdeck.com/hhamon/applying-design-patterns-to-symfony>.

**Horrutiner, César Antonio González.** 2017. Infomed Instituciones. [En línea] 16 de Abril de 2017. [https://es.Gesti%C3%B3n\\_de\\_la\\_informaci%C3%B3n](https://es.Gesti%C3%B3n_de_la_informaci%C3%B3n).

**Iglesias, Raquel Amarilla y Ruesta, Carlota Bustelo.** 2001. *Gestión del conocimiento y Gestión de la Información.* Madrid : s.n., 2001.

**Informática y Tecnología.** 2017. [En línea] 15 de Octubre de 2017. <https://tecnologia-informatica.com/lenguaje-de-programacion/>.

**Ingeniería de Software.** 2013. [En línea] 2013. <https://sites.google.com/site/ingenierialeosw/unidad-1-fundamentos-de-ingenieria-de-software/1-5-definicion-e-historia-de-las-herramientas-case>.

**Ingeniería DRIC. 2018.** ServiceDesk Plus. [En línea] iDric, 2018. [Citado el: 25 de Octubre de 2018.] <http://www.idric.com.mx/service-desk/>.

**ISO/IEC/IEEE INTERNATIONAL STANDARD (EEUU).** ISO/IEC/IEEE 24765, of. 2017: Systems and software engineering-Vocabulary. New York, agosto 28 de 2017.

**Jimenez, Jose. 2015.** Prezi. [En línea] 19 de Marzo de 2015. <https://prezi.com/t1qkd39wvduw/lenguajes-de-programacion-de-lado-del-cliente/>.

**Julián Pérez Porto, María Merino. 2015.** Definicion.de. [En línea] 2015. <https://definicion.de/problemas-de-investigacion/>.

**López, Edwin. 2012.** Portafolio NetBeans. [En línea] 28 de Abril de 2012. <https://sites.google.com/site/portafolionetbeans/que-es-netbeans>.

**Lourdes B. Alpízar Caballero, Haile Trutié Rodríguez. 2015.** *Sistema de Información para la Gestión de Ciencia, Tecnología e Innovación en las Facultades de Ciencias Médicas*. 2015, Revista Cubana de Medicina Militar.

**Martínez, Erick.** Lenguajes de Programación del lado servidor [En línea]. 2014 [Fecha de consulta: 7 noviembre 2018]. Disponible en: HYPERLINK "<http://michelletorres.mx/lenguajes-de-programacion-del-lado-servidor/>" \h <http://michelletorres.mx/lenguajes-de-programacion-del-lado-servidor/> .

**Naranjo, Esteban Roberto de León. 2018.** Módulo para la gestión de servidores proxy squid del sistema Xilema Smart Keeper.

**Naranjo, Francisco José. 2015.** El Blog de Seidor. [En línea] 15 de Enero de 2015. [Citado el: 29 de Octubre de 2018.] [Sistemas de Gestión%20 Valor Estratégico de las Organizaciones.htm](#).

**Nielsen, J.** Usability Engineering. 1993. [En línea]. Consulta: 5 mayo 2018. Disponible en: <https://www.nngroup.com/books/usability-engineering>.

**ORACLE.** Getting Started with MySQL [En línea]. Oracle Corporation, 2017 [Fecha de consulta: 22 octubre 2018]. Disponible en: HYPERLINK "<https://dev.mysql.com/doc/mysql-getting-started/en/>" \h <https://dev.mysql.com/doc/mysql-getting-started/en/>



**Pacheco, Nacho.** Bases de datos y Doctrine [En línea]. 2013 [Fecha de consulta: 14 noviembre 2018]. Disponible en: <http://gitnacho.github.io/symfony-docs-es/book/doctrine.html>.

**PHP.** ¿Qué es PHP? [En línea]. PHP Foundation, 2017 [Fecha de consulta: 12 octubre 2018]. Disponible en: [HYPERLINK "http://php.net/manual/es/intro-what-is.php"](http://php.net/manual/es/intro-what-is.php) \h <http://php.net/manual/es/intro-what-is.php>.

**PHP-FIG.** PHP Standards Recommendation [En línea]. PHP Framework Interop Group, 2018 [Fecha de consulta: 20 marzo 2019]. Disponible en: <https://www.php-fig.org/psr/>

**Pressman, Roger.** Ingeniería del Software. Un enfoque práctico. 7ª ed. México, D.F: The MacGraw-Hill Companies Inc., 2010. ISBN: 978-607-15-0314-5.

**Ramírez, Aaron Arenas. 2017.** Gestión del conocimiento. Concepto y explicación. . [En línea] 11 de Enero de 2017. <https://www.gestiopolis.com/gestion-del-conocimiento-concepto-explicacion/>.

**Ramos, Daniel.** Desarrollo de Software: Estimación Requisitos y Análisis. 2.ª ed. IT Campus Academy, 2016. pp. 125. ISBN: 9781530088614.

**Reguant, Mercedes y Torrado, Mercedes.** El método Delphi. Revista d'Innovació i Recerca en Educació. 2016. vol. 9, no. 1. pp. 87-102. ISSN:2013-2255.

**Rodríguez, Alina y Silva, Luis.** Arquitectura de software para el sistema de visualización médica Vismedic. Revista Cubana de Informática Médica. 2016. vol. 8, no. 1. pp. 75-86. ISSN: 1684-1859.

**Rouse, Margaret. 2016.** Tech Target. [En línea] Diciembre de 2016. <https://searchdatacenter.techtarget.com/es/definicion/Servidor-Web>.

**Sanchez, Ileana R. Alfonso. 2016.** *La sociedad de la Iformación, la Sociedad del Conocimiento y Sociedad del Aprendizaje.Referentes en torno a su formación.* 2016, Reflexiones.

**Sánchez, Tamara Rodríguez. 2015.** *Metodología de desarrollo para la Actividad productiva de la UCI.* Habana : s.n., 2015.

**Sommerville, Ian.** Ingeniería del software. 7ª ed. Madrid: Pearson Educación S.A, 2011. ISBN: 84-7829-074-5.

**Techopedia. 2017.** [En línea] 2017. <https://www.techopedia.com/definition/3243/unified-modeling-language-uml>.

**V&v Quality S.A.** [En línea]. 2016. Consulta: 20 abril 2018. Disponible en: <http://vyvquality.com/pruebas-rendimiento>

**Value Coders. 2017.** [En línea] 1 de Marzo de 2017. <https://www.valuecoders.com/blog/technology-and-apps/symfony-vs-laravel-php-framework-choose/>.

**Vázquez, Yusliel García. 2007.** *GestCult. Sistema de gestión de información de actividades culturales.* Habana : s.n., 2007.

**Verdugo, María Angela Torres. 1981.** *El IRESIE. Un Banco de Información sobre Educación.* 1981.

**Visual Paradigm. 2016.** [En línea] 2016. [https://www.visual-paradigm.com/support/documents/vpuserguide/12/13/5963\\_visualparadi.htm](https://www.visual-paradigm.com/support/documents/vpuserguide/12/13/5963_visualparadi.htm)

## Anexos

### Anexo 1. Historias de Usuario

Tabla 18. HU\_3 Buscar Problema de Investigación

Historia de Usuario	
<b>Número:</b> HU_3	<b>Nombre:</b> Buscar Problema de Investigación
<b>Programador responsable:</b> Ana Carla del Real Baró	<b>Iteración asignada:</b> 1
<b>Prioridad:</b> Baja	
<b>Descripción:</b> El sistema debe permitir realizar una búsqueda simple de los problemas de investigación	
<b>Observaciones:</b> El sistema permite filtrar la búsqueda por el nombre del problema de investigación	
<b>Prototipo de interfaz:</b> No aplica	

Tabla 19. HU\_4 Gestionar Línea Científica

Historia de Usuario	
<b>Número:</b> HU_4	<b>Nombre:</b> Gestionar Línea Científica
<b>Programador responsable:</b> Ana Carla del Real Baró	<b>Iteración asignada:</b> 1
<b>Prioridad:</b> Baja	
<b>Descripción:</b> El sistema debe permitir crear, mostrar, editar y eliminar una línea científica	
<b>Observaciones:</b> Al crear y editar una línea científica, esta debe tener un nombre. El jefe de la comisión será el encargado de añadir las líneas científicas al sistema.	

**Prototipo de interfaz:**

Insertar Línea Científica

Nombre

Text3

Insertar

Volver Atrás

Tabla 20. HU\_5 Restaurar Base de Datos

Historia de Usuario	
<b>Número:</b> HU_5	<b>Nombre:</b> Restaurar Base de Datos
<b>Programador responsable:</b> Ana Carla del Real Baró	<b>Iteración asignada:</b> 1
<b>Prioridad:</b> Media	
<b>Descripción:</b> El sistema debe permitir restaurar la base de datos	
<b>Observaciones:</b> Al restaurar la base de datos se debe tener un servidor ip, nombre de usuario, contraseña y la fecha que se desea restaurar. El administrador será el encargado de restaurar la base de datos.	
<b>Prototipo de interfaz:</b>	
<p>Restaurar Base Datos</p> <p>Servidor IP <input type="text"/> Fecha <input type="text"/></p> <p>Nombre de Usuario <input type="text"/></p> <p>Contraseña <input type="text"/></p> <p>Restaurar</p>	

Tabla 21. HU\_6 Salvar Base de Datos

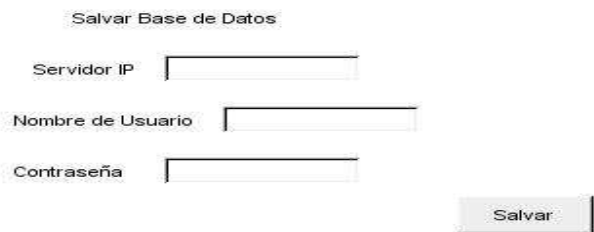
Historia de Usuario	
<b>Número:</b> HU_6	<b>Nombre:</b> Salvar Base de Datos
<b>Programador responsable:</b> Ana Carla del Real Baró	<b>Iteración asignada:</b> 1
<b>Prioridad:</b> Alta	
<b>Descripción:</b> El sistema debe permitir salvar la base de datos	
<b>Observaciones:</b> Al salvar la base de datos se debe tener un servidor ip, nombre de usuario y contraseña. El administrador será el encargado de salvar la base de datos.	
<b>Prototipo de interfaz:</b> 	

Tabla 22. HU\_7 Generar Trazas

Historia de Usuario	
<b>Número:</b> HU_7	<b>Nombre:</b> Generar Trazas por IP
<b>Programador responsable:</b> Ana Carla del Real Baró	<b>Iteración asignada:</b> 1
<b>Prioridad:</b> Media	
<b>Descripción:</b> El sistema debe permitir generar trazas	
<b>Observaciones:</b> El sistema genera las trazas que dejan los usuarios al usar el sistema, estas son mostradas en una tabla con el ip que utilizaron, la fecha, la acción que realizaron y bajo que rol entraron al sistema.	
<b>Prototipo de interfaz:</b> No aplica	

Tabla 23. HU\_10 Generar Reportes

Historia de Usuario	
<b>Número:</b> HU_10	<b>Nombre:</b> Generar Reportes
<b>Programador responsable:</b> Ana Carla del Real Baró	<b>Iteración asignada:</b> 1
<b>Prioridad:</b> Media	
<b>Descripción:</b> El sistema debe permitir generar reportes	
<b>Observaciones:</b> El sistema genera distintos reportes sobre los problemas de investigación ya sean aprobados o resueltos, así como la cantidad de problemas insertados por año. También se generan reportes sobre los problemas asociados a una línea científica o área específica. Estos reportes pueden ser estadísticos o gráficos dependiendo de lo que escoja el usuario.	
<b>Prototipo de interfaz:</b> No aplica	

Tabla 24. HU\_11 Aprobar Problema de Investigación

Historia de Usuario	
<b>Número:</b> HU_11	<b>Nombre:</b> Aprobar Problema de Investigación
<b>Programador responsable:</b> Ana Carla del Real Baró	<b>Iteración asignada:</b> 1
<b>Prioridad:</b> Alta	
<b>Descripción:</b> El sistema debe permitir aprobar un problema de investigación	
<b>Observaciones:</b> El jefe de la comisión es el que tiene los permisos para aprobar un problema de investigación. Es el encargado de revisar los problemas y comprobar que cumplen con los requisitos que debe tener un correcto problema de investigación.	
<b>Prototipo de Interfaz:</b>	
<p style="text-align: center;">Problema de Investigación - Problema UNO</p> <div style="display: flex; justify-content: space-between;"> <div style="width: 45%;"> <p>Título <input type="text" value="Problema UNO"/></p> <p>Línea Científica <input type="text"/></p> <p>Descripción <input type="text" value="TextArea"/></p> </div> <div style="width: 45%;"> <p>Clasificación <input type="text"/></p> <p>Área <input type="text"/></p> </div> </div> <div style="display: flex; justify-content: flex-end; margin-top: 10px;"> <input type="button" value="Regresar"/> <input type="button" value="Aprobar"/> </div>	

Tabla 25. HU\_12 Asignar Problema de Investigación

Historia de Usuario	
<b>Número:</b> HU_12	<b>Nombre:</b> Asignar Problema de Investigación
<b>Programador responsable:</b> Ana Carla del Real Baró	<b>Iteración asignada:</b> 1
<b>Prioridad:</b> Alta	
<b>Descripción:</b> El sistema debe permitir asignar un problema de investigación	
<b>Observaciones:</b> El comité de tesis es el que posee los permisos para realizar esta acción. Un usuario tiene la opción de dar me interesa a un problema de investigación, incluso pueden haber varios usuarios interesados en el mismo problema de investigación. El comité de tesis es el encargado de decidir a que usuario asignarle el problema de investigación para que lo desarrolle posteriormente.	
<p><b>Prototipo de Interfaz:</b></p>	

Tabla 26. HU\_13 Insertar resumen del problema de investigación resuelto

Historia de Usuario	
<b>Número:</b> HU_13	<b>Nombre:</b> Insertar resumen del problema de investigación resuelto

<b>Programador responsable:</b> Ana Carla del Real Baró	<b>Iteración asignada:</b> 1
<b>Prioridad:</b> Media	
<b>Descripción:</b> El sistema debe permitir insertar resumen del problema de investigación resuelto	
<b>Observaciones:</b> El comité de tesis es el encargado de subir el resumen del problema de investigación para que sea mostrado para todos los usuarios en el banco de resultados.	
<b>Prototipo de Interfaz:</b>	
<p>Resumen de Problema de Investigación</p> <div style="border: 1px solid black; padding: 10px; width: fit-content; margin: 0 auto;"> <p>TextArea</p> </div> <div style="display: flex; justify-content: center; gap: 20px; margin-top: 10px;"> <input type="button" value="Volver atrás"/> <input type="button" value="Insertar"/> </div>	

Tabla 27. HU\_14 Mostrar resumen del problema de investigación resuelto

Historia de Usuario	
<b>Número:</b> HU_14	<b>Nombre:</b> Mostrar resumen del problema de investigación resuelto
<b>Programador responsable:</b> Ana Carla del Real Baró	<b>Iteración asignada:</b> 1
<b>Prioridad:</b> Media	
<b>Descripción:</b> El sistema debe permitir mostrar el resumen del problema de investigación resuelto	
<b>Observaciones:</b> El sistema muestra el resumen de los problemas resueltos que se encuentran en el banco de resultados.	
<b>Prototipo de interfaz:</b> No aplica	



Tabla 28. HU\_15 Exportar reportes a pdf

Historia de Usuario	
<b>Número:</b> HU_15	<b>Nombre:</b> Exportar reportes a pdf
<b>Programador responsable:</b> Ana Carla del Real Baró	<b>Iteración asignada:</b> 1
<b>Prioridad:</b> Media	
<b>Descripción:</b> El sistema debe permitir exportar los reportes a pdf	
<b>Observaciones:</b> El sistema exporta solo los reportes estadísticos a pdf.	
<b>Prototipo de interfaz:</b> No aplica	

## Anexo 2. Diagramas de clase de diseño

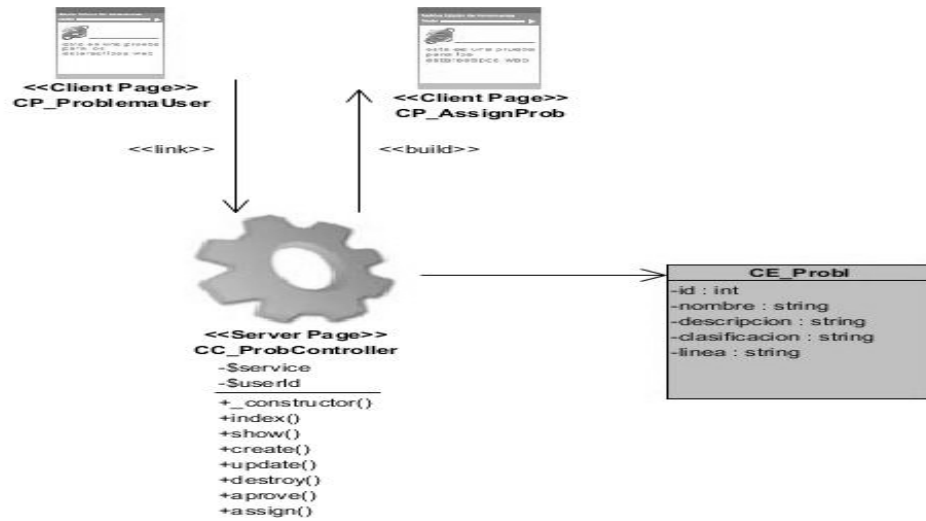


Figura 14. Diagrama de clases del diseño HU Asignar Problema de Investigación (Elaboración propia)

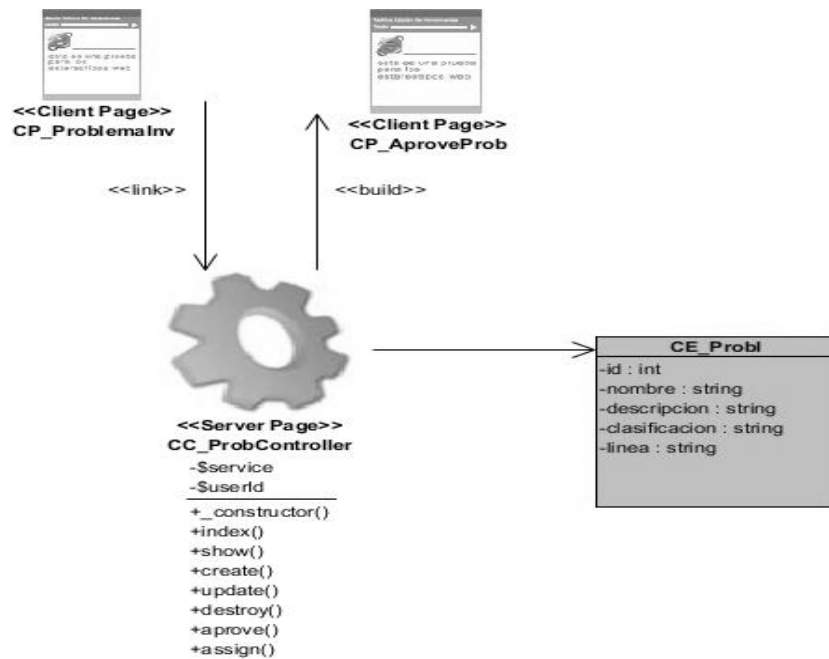


Figura 15. Diagrama de clases del diseño HU Aprobar Problema de Investigación (Elaboración propia)

### Anexo 3. Diagramas de secuencia

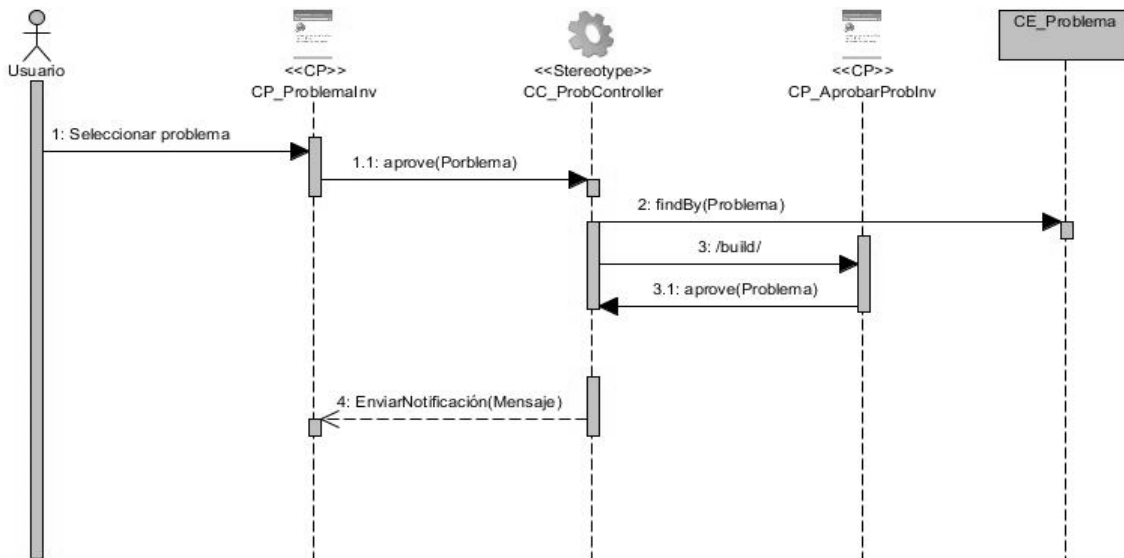


Figura 16. Diagrama de secuencia HU Aprobar Problema de Investigación (Elaboración propia)

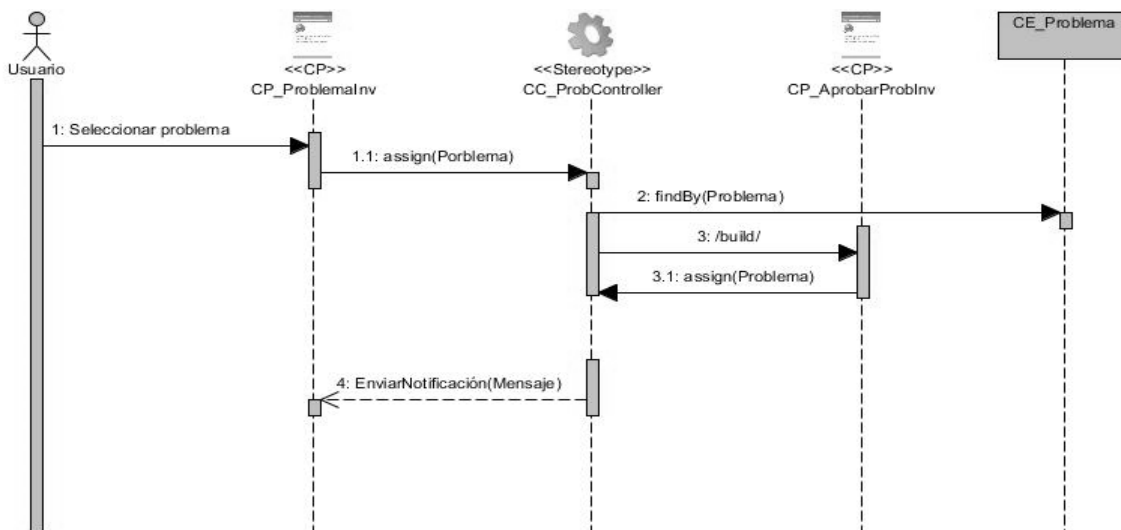


Figura 17. Diagrama de secuencia HU Asignar Problema de Investigación (Elaboración propia)

## Anexo 4. Casos de Prueba

Tabla 29. Caso de Prueba del RF14\_ Aprobar problema de investigación

<b>Caso de Prueba 3: SC RF11_Aprobar problema de investigación</b>			
<b>Condiciones de ejecución:</b> El usuario debe estar autenticado en el sistema y debe estar bajo el rol de jefe de comisión.			
<b>Escenario</b>	<b>Descripción</b>	<b>Respuesta del Sistema</b>	<b>Flujo Central</b>
EC 3.1 Aprobar un problema de investigación	El sistema permite aprobar un problema de investigación de forma correcta	El sistema muestra una notificación al usuario "El problema fue aprobado correctamente"	En el menú superior del sistema: 1. El usuario accede a la opción Problema de Investigación y el sistema muestra una lista de los problemas de investigación registrados. 2. El usuario selecciona la opción aprobar problema de investigación (botón representado por una palomita).
EC 3.2 Aprobar un problema de investigación que ya está aprobado	El sistema no permite aprobar un problema de investigación que ya esté aprobado	El sistema deshabilita el botón que permite aprobar el problema de investigación	

Tabla 30. Caso de Prueba del RF15\_ Asignar problema de investigación

<b>Caso de Prueba 4: SC RF12_Asignar problema de investigación</b>			
<b>Condiciones de ejecución:</b> El usuario debe estar autenticado en el sistema y debe estar bajo el rol de comité de tesis.			
<b>Escenario</b>	<b>Descripción</b>	<b>Respuesta del Sistema</b>	<b>Flujo Central</b>
EC 4.1 Asignar un problema de investigación	El sistema permite aprobar un problema de investigación de forma correcta	El sistema muestra una notificación al usuario "El problema fue asignado a <nombre> correctamente"	En el menú superior del sistema: 1. El usuario accede a la opción Problema Investigación por Usuario y sistema muestra una lista de los problemas de

EC 4.2 Asignar un problema de investigación que ya está asignado	El sistema no permite asignar un problema de investigación que ya esté asignado	El sistema no muestra el problema en la lista de problemas para asignar una vez que ya fue asignado.	investigación que son solicitados por los estudiantes. 2. El usuario selecciona la opción asignar problema de investigación.
--	---	--	---

Tabla 31. Caso de Prueba del RF7\_ Restaurar base de datos

<b>Caso de Prueba 5: RF7_ Restaurar base de datos.</b>			
<b>Condiciones de ejecución:</b> El usuario debe estar autenticado en el sistema y debe haber llenado correctamente el formulario para salvar la base de datos.			
<b>Escenario</b>	<b>Descripción</b>	<b>Respuesta del sistema</b>	<b>Flujo Central</b>
EC 2.1. Restaurar la base de datos de forma correcta	El sistema restaura la base de datos de forma correcta, luego de haber validado los campos del formulario para restaurar la misma.	Muestra un mensaje de confirmación: "La base de datos fue restaurada correctamente".	1. El administrador accede a la opción Restaurar base de datos. 2. El sistema muestra un formulario para restaurar la base de datos. 3. El usuario introduce valores válidos y presiona el botón Restaurar.

## Anexo 5. Encuesta para la identificación de posibles expertos

Este material constituye un instrumento para la investigación del trabajo de diploma en opción al título de Ingeniero en Ciencias Informáticas: **Sistema para la gestión del banco de problemas científicos de la Facultad 1** de la autora: **Ana Carla del Real Baró**, para el Centro de Ideoinformática (CIDI) de la Facultad 1. Usted ha sido identificado como posible experto en el tema y su colaboración es importante para la investigación.

1.- Nombre y Apellidos: \_\_\_\_\_

2.- Cargo que ocupa: \_\_\_\_\_

3.- Años de experiencia: \_\_\_\_

4.- Marque con una cruz (X):

Nivel Escolar: \_\_\_\_ Técnico Medio \_\_\_\_ Enseñanza Media Superior \_\_\_\_ Enseñanza Superior

Grado Científico: \_\_\_\_ Master en Ciencias \_\_\_\_ Doctor en Ciencias \_\_\_\_ Ninguno

5.- Valore en la siguiente tabla con una escala del 1 al 10, el grado de conocimiento que usted posee sobre cada uno de los temas. Considere que la escala es ascendente donde el 1 es que no posee ningún conocimiento y 10 posee el máximo conocimiento del tema:

Tema	Escala de conocimiento
Metodología de la investigación científica	
Gestión del conocimiento	
Gestión de la información	

6.- Realice una autoevaluación del grado de influencia que cada una de las fuentes, que se le presenta a continuación, ha tenido en su conocimiento y criterio para la gestión de problemas científicos. Para ello marque con una cruz (X) según corresponda:

<b>Fuentes de Argumentación</b>	<b>Alto</b>	<b>Medio</b>	<b>Bajo</b>
Análisis teóricos realizados por usted			
Experiencia obtenida			
Trabajos de autores nacionales			
Trabajos de autores extranjeros			
Su conocimiento del estado del problema en el extranjero			
Su Intuición			

Gracias por su colaboración.

## Anexo 6. Encuesta a expertos

Co.:

Usted fue seleccionado como miembro del panel de experto para validar la **hipótesis científica**: El desarrollo de un sistema para la gestión del banco de problemas científicos para la Facultad 1 mejora el proceso de gestión de problemas de investigación en el Vicedecanato de Investigación y Postgrado de la Facultad 1. En la aplicación del modelo matemático Torgerson del método Delphi, se le solicita que responda el siguiente cuestionario según su nivel de valoración.

Respecto al Sistema para la gestión del banco de problemas científicos de la Facultad 1 emita una valoración según las siguientes categorías de evaluación: **MA** si considera que la sentencia es Muy Adecuada, **A** si la considera adecuada, **PA** si la valora Poco Adecuada e **I** si es Inadecuada. Marque con una cruz (X):

1. Facilidad en la gestión de los problemas de investigación.
2. Facilidad en la asignación de temas de tesis.
3. Rapidez en la respuesta de peticiones.
4. Presentación de una interfaz agradable e intuitiva para el usuario.

Nº	Sentencia	Categorías Evaluativas			
		MA	A	PA	I
1	Facilidad en la gestión de los problemas de investigación.				
2	Facilidad en la asignación de temas de tesis.				
3	Rapidez en la respuesta de peticiones.				
4	Presentación de una interfaz agradable e intuitiva para el usuario.				

Gracias por su colaboración.

Atentamente: Ana Carla del Real Baró.