



**UNIVERSIDAD DE LAS CIENCIAS INFORMÁTICAS**

**FACULTAD 5**

**CENTRO DE CONSULTORÍA Y DESARROLLO DE ARQUITECTURAS  
EMPRESARIALES**

**Portal para la gestión de la información asociada a la  
actividad científica del Ministerio de Educación Superior**

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

**Autores:** Yessida Tamayo Barrio  
Raúl Pérez Guevara

**Tutor:** Dr.C. Orestes Febles Díaz

**La Habana, junio de 2016**

## DECLARACIÓN JURADA DE AUTORÍA

Declaro ser autor de la presente tesis y se reconoce a la Universidad de las Ciencias Informáticas los derechos patrimoniales de la misma, con carácter exclusivo.

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

Nombre de los autores

Yessida Tamayo Barrio      Raúl Pérez Guevara

---

Nombre del tutor principal

Dr.C. Orestes Febles Díaz

---

## **AGRADECIMIENTOS**

A nuestros padres por el apoyo incondicional, por ser la inspiración de todo el trabajo realizado, por ayudarnos en todo, incluso cuando se tenían que privar de sus propias necesidades anteponiendo las de nosotros.

A mi hermana Dayessi por ser amiga y apoyo en todo momento.

A mi primo Orlando por ser como un hermano para mí, y un amigo en la escuela.

A Orestes por su papel fundamental como tutor, y dedicarnos el tiempo para que los resultados fuesen satisfactorios.

A Armando Maso Mosqueda, por la atención que nos prestó a pesar de no ser nuestro tutor para ayudarnos en las dudas que nos surgieron.

A Yurien Santos Valdés por la ayuda prestada, siendo vital para el desarrollo de nuestro sistema.

A nuestro jurado que con sus acotaciones en la predefensa nos fueron de gran ayuda.

## **DEDICATORIA**

A todos aquellos que de una manera u otra estuvieron todo este tiempo junto a nosotros, a nuestros padres, familiares, compañeros del aula, a Lidia María, Brian, Claudia, Yaima, Odette, Liliana, Laura y demás amigos que se han ganado nuestro cariño y respeto.

## RESUMEN

Actualmente la información y el conocimiento asociado a la actividad científica del Ministerio de Educación Superior carece de una plataforma para su gestión. La Dirección de Ciencia y Técnica del mismo es la encargada de manejar toda esta información y los procesos que conlleva, acciones que realiza hoy de manera manual. Entre las insuficiencias que hoy está presentando esta dirección se encuentran la gestión de los proyectos, ya que, para conocer el estado de los mismos, si están avalados o no, entre otros datos de interés, es necesaria una interacción por vía telefónica, por otro lado, existe escasa divulgación de los proyectos investigativos acompañado de una carencia de mecanismos de interacción que logren mantener una correcta gestión del conocimiento. Además, no se tiene una forma organizada de dar a conocer las principales tareas (reuniones, u otras actividades) de forma tal que se hagan extensibles para todos los involucrados. Para darle solución a dicha problemática el equipo de trabajo ha desarrollado un sistema para la gestión de la investigación asociada a la actividad científica del Ministerio de Educación Superior. En su construcción se emplearon tecnologías tales como Spring, JSF (*Java Server Faces*), Primefaces, Hibernate y PostgreSQL y se utilizó el Proceso Unificado Ágil como metodología de desarrollo. Las pruebas al producto obtenido garantizaron la confiabilidad del mismo y por tanto la entrega de una solución útil, la que con su empleo fortalece sus mecanismos de gestión.

**Palabras claves:** gestión, actividad científica, investigaciones, portal.

# CONTENIDOS

<b>INTRODUCCIÓN .....</b>	<b>1</b>
<b>CAPÍTULO 1 Marco teórico y referencial de la investigación .....</b>	<b>4</b>
1.1 Web 2.0 .....	4
1.2 <i>CMS Content Manager System</i> (Sistema Gestor de Contenidos) .....	5
1.3 Portales web.....	7
1.3.1 <i>Portlets</i> .....	10
1.3.2 Marco de trabajo para el desarrollo de portales .....	11
1.3.3 Herramientas del Liferay .....	15
1.4. Metodologías.....	17
<b>CAPÍTULO 2 Planificación, diseño y desarrollo del sistema .....</b>	<b>20</b>
2.1 Características del portal .....	20
2.2 Modelo de dominio .....	24
2.3 Arquitectura del portal.....	25
2.3.1 Estilos y patrones arquitectónico utilizados .....	26
2.3.2 Patrones de diseño. ....	26
2.4 Historias de usuario.....	29
2.4.1 Descripción de las HU.....	29
2.4.2 Modelo de diseño.....	39
2.4.3 Arquitectura de la información.....	41
<b>CAPÍTULO 3 Implementación y pruebas .....</b>	<b>49</b>
3.1 Estándar de código.....	49
3.2 Despliegue del Liferay .....	50
3.3 Pruebas.....	52
3.4 Estrategia de pruebas.....	52

3.5 Casos de prueba.....	56
3.6 Lista de chequeos.....	56
3.7 Resultados.....	57
3.8 Pruebas de aceptación.....	57
3.9 Pruebas de rendimiento.....	60
<b>CONCLUSIONES.....</b>	<b>64</b>
<b>RECOMENDACIONES.....</b>	<b>65</b>
<b>REFERENCIAS BIBLIOGRÁFICAS.....</b>	<b>66</b>
<b>ANEXOS.....</b>	<b>69</b>

## **INTRODUCCIÓN**

La evolución constante de las tecnologías ha contribuido en gran medida a la informatización de los principales procesos de las organizaciones. Las entidades del gobierno cubano están inmersas en una fase continua de informatización de sus procesos claves. Desde muy temprano han identificado la necesidad de dominar e introducir las tecnologías de la información y las comunicaciones para agilizar cada uno de los procedimientos realizados en sus instituciones. Para llevar a cabo esto se diseñó un programa rector para la informatización de la sociedad cubana promoviendo el uso masivo de las tecnologías para involucrar los sectores de la sociedad.

El Ministerio de Educación Superior (MES) es un sistema integrado por instituciones de educación superior y Entidades de Ciencia e Innovación Tecnológica (ECTI), que rige metodológicamente la educación superior cubana. Este sistema, bajo la dirección del Partido Comunista de Cuba (PCC) y el Gobierno, forma y consolida valores patrios y de profundo sentido humanista, a la par que preserva, genera y promueve conocimientos, habilidades y competencias, que se reflejan en la formación integral del profesional, la educación posgraduada, las actividades de ciencia, tecnología e innovación y la extensión, con pertinencia, actualidad, eficacia, eficiencia y racionalidad, acorde con las exigencias de la sociedad (MES, 2015).

El MES está compuesto por varias direcciones, entre ellos se encuentran:

- Dirección de Recursos Humanos.
- Dirección de Informatización.
- Dirección de Ciencia y Técnica (DCT).

El MES no está ajeno a estos procesos de informatización, es una de las organizaciones que presta gran interés en relación al desarrollo de software para la mejora de sus procesos. Entre las instituciones que la componen se encuentran la Universidad de La Habana (UH), Instituto Superior de Diseño Industrial (ISDI), la Universidad de Ciencias Informáticas (UCI), Universidad de Pinar del Río (UPR), Instituto Superior Politécnico José Antonio Echeverría (ISPJAE) y muchas más, las cuales tienen una activa vida científica por lo que necesitan ser gestionadas de manera eficiente.

La Dirección de Ciencia y Técnica del MES ha detectado un conjunto de insuficiencias que atentan contra la efectividad en la gestión de la actividad científica. Dentro de las más importantes se encuentran las siguientes:

- Actualmente los proyectos de investigación que se llevan a cabo en todas las instituciones pertenecientes al MES se manejan y controlan de manera manual.
- Escasa divulgación de la actividad científica de las universidades y ECTI.
- No se puede realizar una actualización constante de los registros de los proyectos avalados.

- Es costoso conocer el estado en que se encuentra cada uno de los proyectos investigativos.
- Se hace necesario un grupo de reportes que apoyen la toma de decisiones en la dirección del ministerio.
- Existen insuficiencias en cuanto a la disponibilidad de la información.
- Carencia de mecanismos de interacción que logren mantener una correcta gestión del conocimiento.
- No se tiene una forma organizada de dar a conocer las principales tareas (reuniones, u otras actividades) de forma tal que se hagan extensibles para todos los involucrados.

Para contribuir a solucionar la problemática anteriormente descrita, esta dirección del MES ha propuesto un conjunto de acciones. Dentro de ellas, el desarrollo de un sistema informático que facilite la gestión de la información y el conocimiento asociado a la actividad científica de las universidades y Entidades de Ciencia, Tecnología e Innovación (ECTI), adscriptas al MES para contribuir a la colaboración entre investigadores, con lo que se podrá supervisar desde el MES, el estado de la ciencia, así como en qué ramas se investiga.

La Universidad de Ciencias Informáticas (UCI) fue concebida como un centro de nuevo tipo, de alcance nacional, de características atípicas y tareas concretas en el proyecto de informatización de la sociedad cubana, con énfasis en la producción de software. Esta juega un papel importante en la informatización de los procesos de la sociedad cubana. La misma ha desarrollado numerosos sistemas para empresas tanto nacionales como internacionales, permitiéndole la adquisición de experiencias y buenas prácticas. Por tal motivo ha sido llamada a solucionar los problemas que presenta la Dirección de Ciencia y Técnica (DCT) del MES.

De lo anteriormente expuesto se identificó el siguiente **problema a resolver**: ¿cómo influir favorablemente en la gestión de la información y el conocimiento asociado a la actividad científica del Ministerio de Educación Superior?

En esta investigación se presenta como **objeto de estudio** el proceso de gestión de la información asociada a la actividad científica del MES; y teniendo como **campo de acción**: las plataformas para el desarrollo y despliegue de portales empresariales.

Para resolver el problema anterior se planteó como **objetivo general** de este trabajo: desarrollar un sistema que permita gestionar la actividad científica del MES. El objetivo general se desglosa en los siguientes **objetivos específicos**:

- Elaborar el marco teórico y referencial asociado a portales empresariales y sistemas relacionados con la gestión de la información y el conocimiento.
- Diagnosticar las necesidades de información en la Dirección de Ciencia y Técnica del Ministerio de Educación Superior.
- Implementar un sistema para la gestión de la Información de la actividad científica del MES.
- Validar el sistema usando técnicas y herramientas asociadas a la ingeniería de software.
- Proponer arquitectura de despliegue para un escenario de alta disponibilidad.

En el desarrollo del sistema se utilizaron los siguientes **métodos científicos**:

**Métodos teóricos:**

- **Analítico - sintético:** para descomponer el problema de investigación en elementos, profundizar en su estudio y luego sintetizarlos en la solución propuesta.
- **Modelación:** en el diseño de la arquitectura del sistema y de cada una de las vistas que la conforman, así como para las bases de datos para que persista la información.
- **Sistémico:** para integrar armónicamente los componentes diseñados en la solución propuesta.

**Métodos empíricos:**

- **Análisis documental:** en la revisión de la literatura especializada, tanto académica como empresarial, para extraer la información necesaria que permitió realizar el proceso de investigación.
- **Entrevistas en profundidad:** para refinar el diseño de la solución propuesta.
- **Pruebas de calidad:** para evaluar la calidad de la solución desarrollada.
- **Controles de aplicabilidad:** para validar con los profesionales la aplicabilidad de la solución desarrollada.

Con la implementación de este sistema se busca lograr los siguientes beneficios: permitir homogeneizar la información que se maneja en los proyectos, agilizar el trabajo y lograr una centralización de la información de los mismos.

En general el apoyo en la toma de decisiones, el almacenamiento, la transferencia y publicación de conocimientos e informaciones. Por otra parte, el aprovechamiento de los beneficios que aporta el trabajo con la web 2.0 donde los usuarios son parte del sitio web aportando opiniones, comentarios, introduciendo nuevo contenido, en general enriqueciendo y siendo parte del portal. La tesis está estructurada en introducción, tres capítulos, conclusiones, recomendaciones, referencias bibliográficas y un cuerpo de anexos. Los capítulos van a estar distribuidos de la siguiente forma:

- **Capítulo 1:** contiene el marco teórico y referencial de la investigación, relacionado con el portal web para la gestión de la información asociada a la actividad científica del MES y las herramientas informáticas apropiadas para su desarrollo.
- **Capítulo 2:** aborda la concepción de la solución informática propuesta en términos de su diseño arquitectónico, así como la metodología a utilizar y los artefactos que la misma genera.
- **Capítulo 3:** contiene el diseño y el resultado de las pruebas de calidad aplicadas a los componentes desarrollados, así como el esquema de despliegue.

## CAPÍTULO 1 Marco teórico y referencial de la investigación

En la presente sección se expone una solución técnica, sin entrar en detalles de diseño e implementación, que responda al problema definido. Para ello, se describen las herramientas a utilizar en el desarrollo de la solución, clasificadas por el papel que desempeñan dentro del sistema.

### 1.1 Web 2.0

Al igual que la revolución industrial transformó radicalmente la manera de producir, intercambiar y promocionar los productos y servicios que ofrecía, la revolución digital está siendo hoy un desafío para las empresas que tienen como perspectiva realizar un cambio en sus negocios y en la forma de manejarlos.

Las aplicaciones web 2.0 son aquellas que sacan partido a las ventajas intrínsecas de la web, ofreciendo un servicio continuamente actualizado que mejora cuanto más gente lo use, utilizando y remezclando los datos de múltiples recursos, incluyendo los usuarios individuales, a la vez que ofrecen sus propios datos y servicios de tal forma que pueden ser reutilizados por otros, creando una “arquitectura de participación” en red, yendo más allá de la página de la web 1.0 para ofrecer experiencias de usuario cada vez más ricas (ARNAL, 2007).

Cuando se habla de web 2.0 se está haciendo referencia al uso de determinadas tecnologías (*ajax*, *mashups*, software social, rss) y a la aplicación de una determinada actitud en el diseño de servicios web.

Ejemplos de servicios web 2.0:

- Amazon.
- Wikipedia.
- Google Maps.
- Del.icio.us.

La web 2.0 permite desarrollar una inteligencia colectiva donde cada uno de los que interactúan puede colaborar. Ha permitido el desarrollo y la puesta en práctica de ideas y conceptos teóricos que internet no soportaba hasta el momento. Propone nuevas formas de producción y de edición de contenidos. Entre las ventajas que el trabajo con esta web brinda se encuentran la posibilidad de asignar diferentes permisos a los usuarios. Contiene mecanismos que facilitan la protección contra acceso ilegal permitiendo realizarle diferentes tipos de auditorías. Por otro lado, las herramientas de la web 2.0 son fáciles de usar necesitando poco tiempo para su aprendizaje.

Permite que los usuarios enriquezcan el contenido y encuentre en el sitio web un punto de referencia para intercambiar ideas y conocimientos a través de foros, blogs y wikis. Con los blogs se puede a través de un sistema simple y económico exponer ideas y redefinir conceptos, también de publicitar, mientras que a través de las wikis permiten a los usuarios crear, investigar temas y redefinir conceptos, *online*, de modo colaborativo en páginas conceptuales, realzar

/informar sobre la ejecución de proyectos, seguir la metodología de “aprendizaje basado en problemas”, para organizar reuniones. Es decir que sus múltiples aportes para educación. Con los foros por su parte, los usuarios pueden compartir opiniones, experiencias y dudas sobre un tema en particular, se puede expresar y responder opiniones. Se logra que a partir de un tema diferentes personas puedan dar su opinión, y así fomentar la interacción entre ellos. Aunque el usuario entre después de iniciado un tema puede entenderlo de manera rápida uniéndose a la discusión.



Figura 1.1 Mapa visual de la web 2.0 (COMMONS, 2007).

## 1.2 CMS Content Manager System (Sistema Gestor de Contenidos)

Un Sistema de Gestión de Contenido es un software que permite la creación y administración de los contenidos de una página web, principalmente, de forma automática. Así, con él podemos publicar, editar, borrar, otorgar permisos de acceso o establecer los módulos visibles para el visitante final de la página. El CMS está formado por 2 elementos:

- La aplicación gestora de contenidos (CMA): el elemento CMA permite al gestor de contenidos o autor realizar la creación, modificación y eliminación de contenido en un sitio web sin necesidad de tener conocimientos de lenguaje HTML.
- La aplicación dispensadora de contenidos (CDA): el CDA usa y compila la información para actualizar el sitio web.

En definitiva, cuando hablamos de un CMS nos referimos a una interfaz que controla una o varias bases de datos donde se aloja el contenido del sitio. Las características de un CMS pueden variar, pero la mayoría incluye publicación basada en web, indexación, revisión, búsqueda y recuperación de la información (MAYOR, 2014).

### Ejemplos de CMS:

- **WordPress** es el CMS más popular en la actualidad. Te permite configurar fácilmente blogs y sitios web utilizando MySQL y PHP. Su curva de aprendizaje es muy rápida. Si quieres crear tu propio sitio web gratuito, rápido y de manera muy fácil, WordPress es la mejor solución (BENITO, 2016).
- **Drupal** es una plataforma de gestión de contenido de código abierto que ha ayudado a crear millones de sitios web. Tiene una comunidad muy activa, pero su curva de aprendizaje es algo mayor que en el caso de WordPress. Para usuarios más experimentados. Puedes utilizar Drupal para crear desde blogs personales a páginas web empresariales. Dispone de miles de módulos adicionales y diseños. Requiere PHP y MYSQL (BENITO, 2016).
- **Joomla** es un sistema de gestión de contenido que te permite construir sitios web y blogs. Con una curva de aprendizaje situada entre WordPress y Drupal. Una de las opciones más populares junto a WordPress y Drupal (BENITO, 2016).

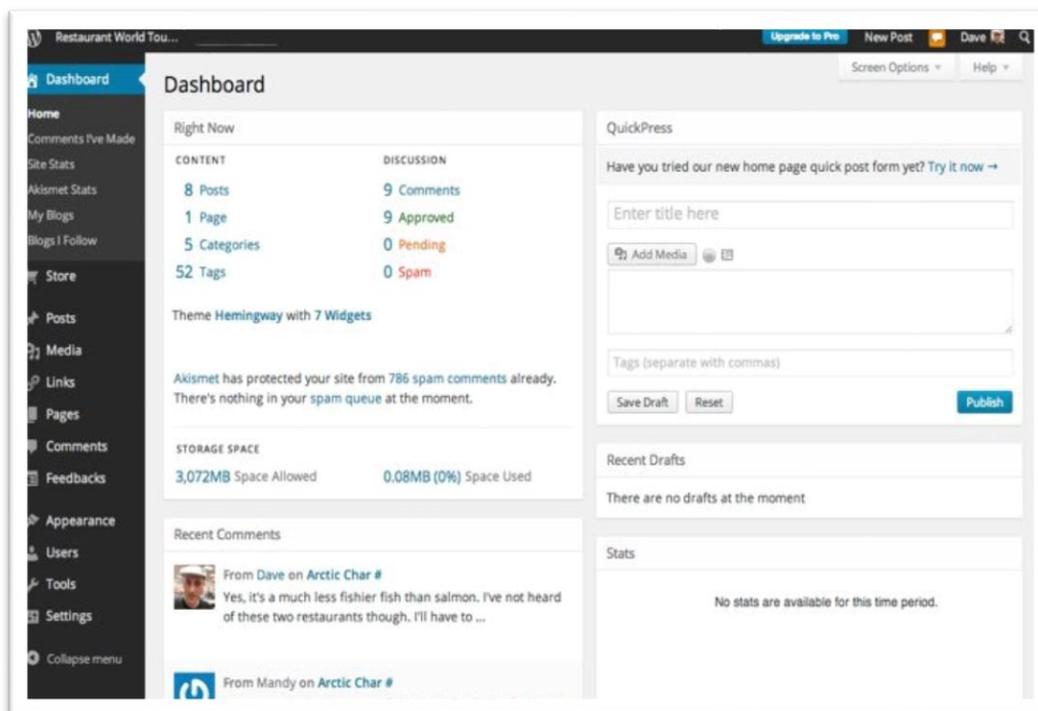


Figura 1.2 WordPress: ejemplo de CMS (BENITO, 2016).

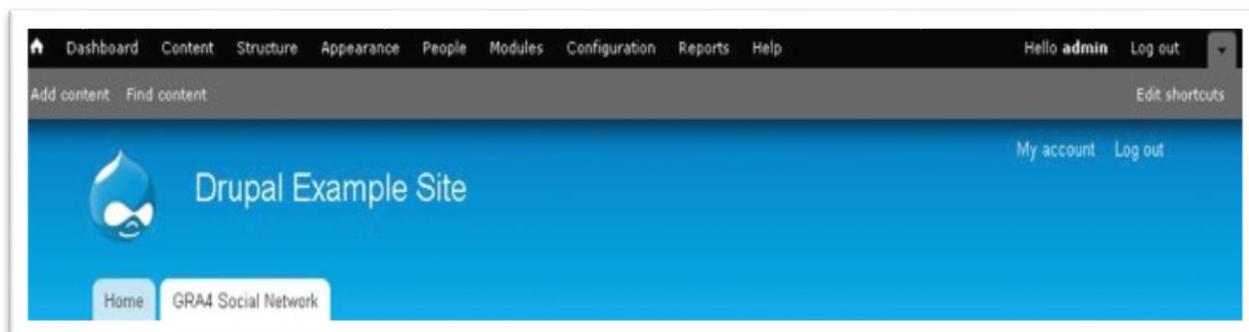


Figura 1.3 Drupal: ejemplo de CMS (BENITO, 2016).

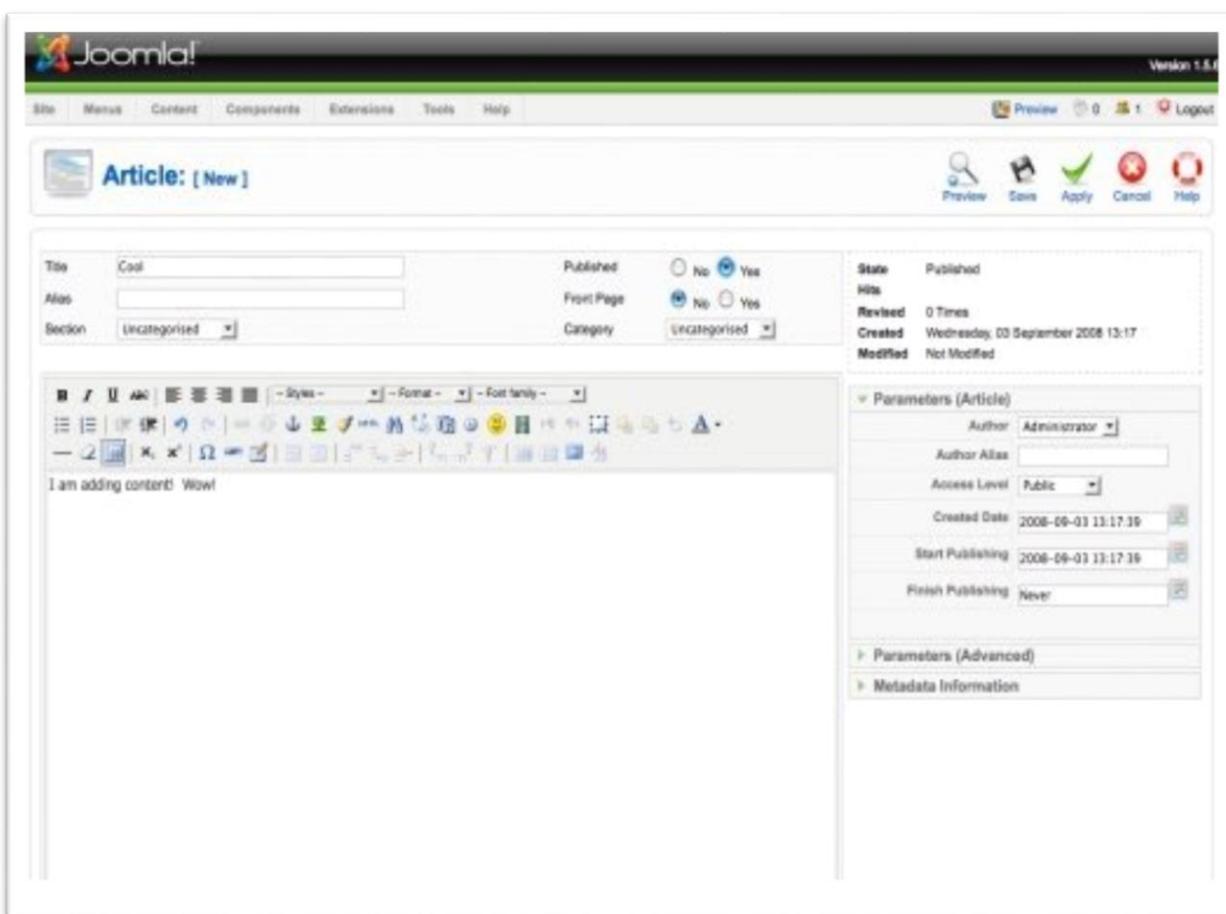


Figura 1.4 Joomla: ejemplo de CMS (BENITO, 2016).

### 1.3 Portales web

Es un punto de entrada a internet donde se organizan sus contenidos, ayudando al usuario y concentrando servicios y productos de forma que le permitan realizar cuanto necesita hacer en

la red a diario, o al menos que pueda encontrar allí todo cuanto utiliza cotidianamente sin necesidad de salir de dicho sitio (GÓMEZ, 2001).

**Beneficios:**

- Optimiza los recursos informáticos de la empresa. Ayuda a reducir la redundancia de datos en los sistemas. Optimiza los recursos informáticos, transparentando la ubicación de los sistemas para el usuario. Proporciona una plataforma única e integrada para administrar las aplicaciones intranet, extranet e internet. Permite que los usuarios tomen decisiones, mostrando la información empresarial fundamental en una ubicación central.
- Reduce tareas redundantes. Proporciona una experiencia fácil, familiar y estándar para el usuario, reduciendo la necesidad de capacitación. Simplifica el acceso a la información de toda la organización, tanto estructurada como sin estructurar, mediante distintos módulos.
- Mejora la comunicación y el enfoque de los usuarios en sus tareas cotidianas. Proporciona mecanismos para potenciar la productividad, simplificando las actividades empresariales diarias, mediante el uso de los flujos de trabajo que ayuden a definir prioridades, realizan tareas automatizadas y generan notificaciones o alertas a los usuarios cuando la información es crítica. Conduce a los usuarios hacia la información que está orientada a sus funciones dentro de la empresa, por lo que reduce los tiempos de desatención.
- Reduce gastos en las Tecnologías de Información. Simplifica la administración de los sistemas, ya que se reducen las plataformas que se tienen que soportar, así como las modificaciones y mantenimiento, debido a que el enfoque del portal es por servicios, por lo que el impacto a las modificaciones se realiza en el servicio que lo requiere, sin afectar los otros procesos.
- Mejora la seguridad de la información. Proporciona un modelo unificado de seguridad que permite reducir redundancias en las administraciones de cuentas de acceso y proporciona seguridad normalizada a las aplicaciones, por lo que elimina el aislamiento de la información de la empresa.
- Garantiza la interconectividad de la información. Reduce la información redundante resolviendo los problemas de desincronización de la misma y permite compartirla, conectando a los usuarios con ésta y los conocimientos. Tiene mecanismos que permiten administrar y volver a usar el contenido para incrementar su valor en la empresa (D, 2007).

**Tipos de portales:**

- Generales (orientados a toda la población).
- Verticales (provee de información y servicios a un sector en particular, con contenidos concretos y centrados en un tema.).
- Corporativos o Empresariales (destinados a las personas relacionadas con empresas o instituciones).

El sistema de gestión de la información asociada a la actividad científica comparte características de los portales empresariales ya que el mismo proveerá de información de la empresa a los trabajadores, se podrá acceder a datos de la propia institución, la edición del trabajo propio, el contacto con investigaciones de otros usuarios, entre otros. Entre los resultados más útiles que brinda este tipo de portal se encuentran la automatización de los procesos, el enlace de información de diferentes fuentes y la integración de nuevos servicios o aplicaciones según las necesidades crecientes.

Para el desarrollo de un portal web se necesita un equipo preparado, así como una adecuada planificación, es preciso tener en cuenta aspectos como el público objetivo, necesidades e intereses y de qué modo pueden quedar satisfechos. Al plantearse los contenidos a incluir y el tratamiento que se le va a dar, evidentemente hay que tener en cuenta al usuario. Como una alternativa en la construcción de portales destaca el empleo de los **portlets**. Estos componentes son alojados en el portal y pueden mostrar extractos de sitios web, ejecutar búsquedas o acceder a colecciones de información.

#### Ejemplos de portales empresariales:

- **Clinked:** gestión de proyectos en línea, el portal del cliente, trabajo en equipo y plataforma de colaboración que permite a las personas interactuar dondequiera que se encuentren (CAPTERRA, 2016).
- **Bitrix24:** solución de gama alta diseñada para la colaboración eficaz, la comunicación, las redes sociales, y el flujo de trabajo y gestión del conocimiento (CAPTERRA, 2016).
- **eXo Platform:** software de colaboración social de código abierto diseñado para empresas sobre la base de estándares, extensible y con diseño agradable (CAPTERRA, 2016).

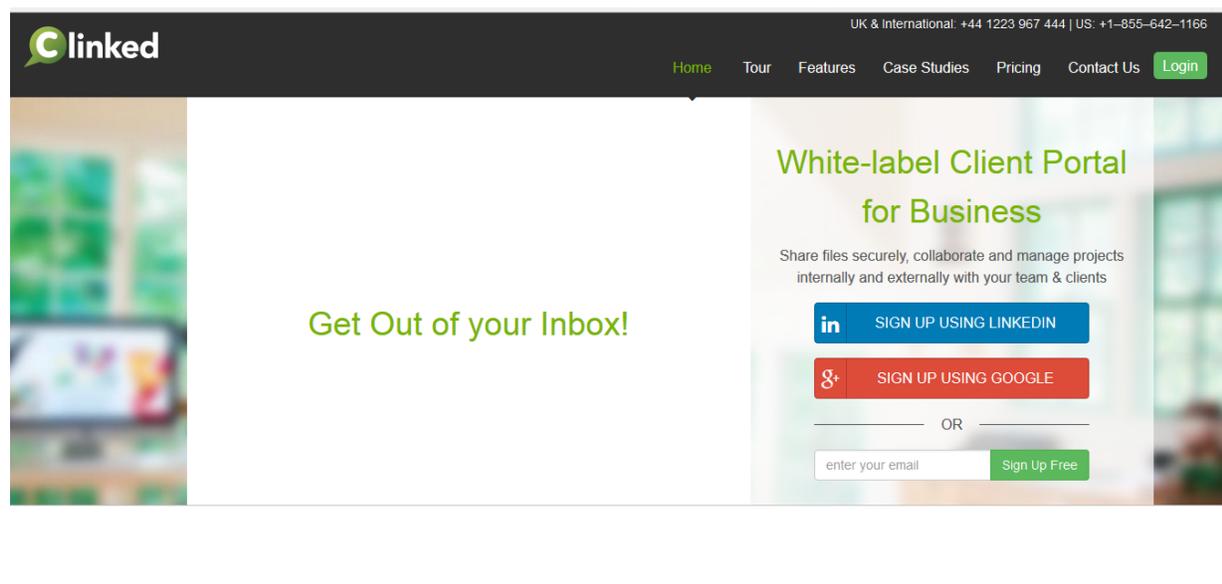


Figura 1.5 Ejemplo de portal empresarial: Clinked (CAPTERRA, 2016).

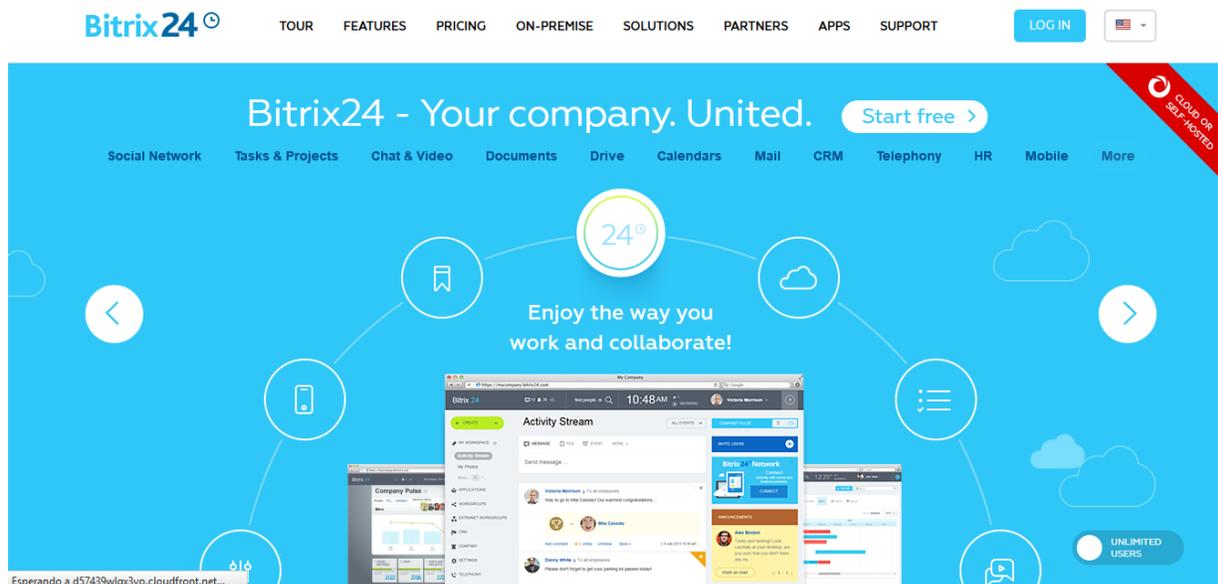


Figura 1.6 Ejemplo de portal empresarial: Bitrix24 (CAPTERRA, 2016).

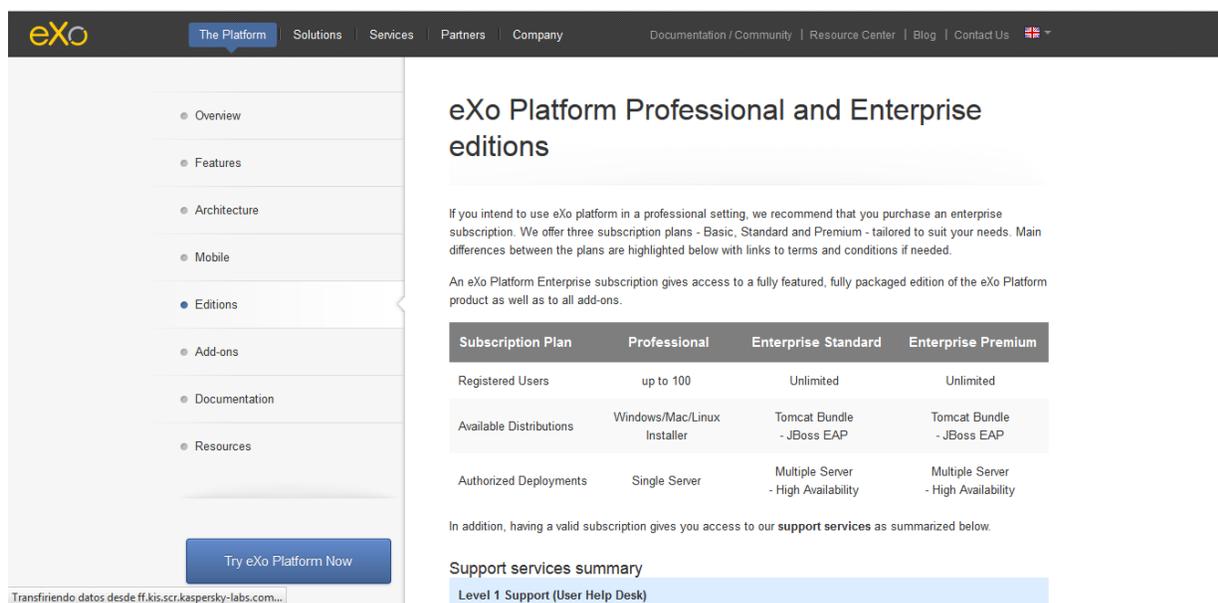


Figura 1.7 Ejemplo de portal empresarial: eXo (CAPTERRA, 2016).

### 1.3.1 Portlets

Los *portlets* son aplicaciones dentro de un portal de forma similar a como los *servlets* son aplicaciones en un servidor web. La diferencia entre ambos es que los *portlets* son aplicaciones completas que incluyen también la interfaz de usuario. Son muy semejantes a las aplicaciones de escritorio Windows, en el sentido de que un *portlet* devuelve fragmentos de XHTML (o cualquier otro lenguaje de marcado) encuadrados dentro de un marco con distintos controles genéricos (OSCAR DÍAZ, 2006). Entre las ventajas del desarrollo basado en *portlets* están:

- Desarrollo independiente. La independencia de los *portlets* permite desarrollarlos de forma separada, y en paralelo con otros desarrollos. Esta característica acelera el tiempo de implementación.
- Personalización. El aspecto final del portal dependerá de los *portlets* que contenga y éstos pueden ordenarse y cambiar esta ordenación de una manera sencilla.
- Seguridad frente a fallos. Un fallo en un *portlet* no se transmite al resto de la aplicación por lo que el único componente que deja de funcionar es el *portlet* que dio el fallo (OSCAR DÍAZ, 2006).

### Uso de las Aplicaciones Enriquecidas en Internet (RIA) en los portales

RIA es un nuevo tipo de aplicación web cuyo objetivo es incrementar y mejorar las opciones y capacidades de las aplicaciones web tradicionales. Estas aplicaciones son desarrolladas en los lenguajes de marcado propio y ejecutados en servidores propios. Las aplicaciones RIA se basan principalmente en el usuario, aplicación que contiene en su desarrollo de trabajo la arquitectura para obtener efectos eficientes y acortar la brecha entre las aplicaciones web y las de escritorio. RIA involucra interacción, velocidad y flexibilidad; mediante el entorno de RIA, no se produce recargas de páginas ya que desde el inicio se presenta toda la página y los datos de la aplicación y así determinar lo que desea cargar (NIETO y ALVARADO, 2011).

#### 1.3.2 Marco de trabajo para el desarrollo de portales

Hoy se persigue migrar hacia software libres, entre las ventajas del software de código abierto se encuentran que la disponibilidad del código fuente hace posible que usuarios, programadores y empresas se involucren en el desarrollo de aplicaciones. De esta forma, el proceso de detección y corrección de errores se lleva a cabo de forma eficiente, así como la implementación de nuevas características en caso de que sea necesario.

Se puede adaptar a través de modificaciones a los programas según las necesidades de las empresas, no existe un gasto de dinero en la compra de licencias, solo se emplearía gastos en capacitar al personal. Por otra parte, no sería necesario contratar a una empresa específica a la hora de realizar un mantenimiento al sistema, ya que basta cualquiera que tenga la preparación necesaria para realizar dicha tarea. Por tales motivos el equipo de trabajo hizo una selección de las principales herramientas tratando que las mismas sigan los principios de código abierto.

Se utilizó el **Apache Tomcat 7.0** como contenedor de *servlets*. *Apache* es un servidor web de código libre robusto cuya implementación se realiza de forma colaborativa, con prestaciones y funcionalidades equivalentes a las de los servidores comerciales (FOUNDATION, THE APACHE SOFTWARE, 2016a). Existen numerosas plataformas corporativas para la construcción de portales, para el desarrollo del sistema para la gestión de la información asociada a la actividad científica del Ministerio de Educación Superior se utilizó el **Liferay 6.2**.

Para el tratamiento de entornos de desarrollo integrados se usó como plataforma el **Eclipse Mars 4.5.0**, es un entorno de código abierto multiplataforma para desarrollar “aplicaciones de clientes enriquecidos”. Permite construir un entorno a partir de componentes conectados (*plugin*). Proporciona entornos de desarrollo y plataformas para casi todos los idiomas y arquitecturas, es

famoso por favorecer la creación de entornos de desarrollo de escritorio, web y de nube. Esta plataforma ofrece la más amplia colección herramientas complementarias para los desarrolladores de software (FOUNDATION, THE ECLIPSE, 2016c).

**Liferay** entra en la clasificación de gestor de contenidos ya que permite desarrollar tantos sitios web como entornos colaborativos online. Este se sitúa entre los Sistemas de Gestión de Contenidos (CMS) referentes del sector empresarial. Pero se puede afirmar que Liferay es más que un CMS, es un *Framework* para el desarrollo de aplicaciones web formado por más de 60 *portlets*.

Liferay Portal es una plataforma web corporativa que permite crear soluciones empresariales con resultados inmediatos y valor a largo plazo. Es una de las tecnologías líder de portal corporativo *Open Source*, prestigio que ha logrado por sus resultados probados en el mundo real con clientes de reconocidos prestigios en diversos sectores. Gracias a la contribución de los clientes de la plataforma mediante desarrollos patrocinados tiene innovación continua y una nueva versión de la plataforma cada ocho meses.

Se puede afirmar que el Liferay es un paquete “todo en uno” ya que incluye una amplia gama de funcionalidades de producto entre las que están: gestión de contenidos y documentos con integración de *Microsoft Office*, edición web y espacios de trabajos compartidos, colaboración a nivel de empresa (*Enterprise collaboration*), redes sociales o híbridas, y portales corporativos y gestión de identidades (LIFERAY, 2016).

¿Por qué usar Liferay para gestionar el sistema y no un CMS?

Existe una serie de situaciones desde el punto de vista del usuario final en las cuales contar con todos los recursos que proporciona Liferay resulta una mejor opción en comparación con el desarrollo de un sitio web a medida o usar un software CMS con diferentes características y orientaciones (ROSAS, 2016).

La instalación por defecto de varios sistemas populares de gestión de contenidos incluye plantillas de páginas principales donde se prioriza mostrar contenido en un listado ordenado cronológicamente, a manera de un blog o de un sitio de noticias, y que requiere mayor esfuerzo de desarrollo para conseguir una plantilla con una portada distinta (ROSAS, 2016).

Por otra parte, Liferay usa un enfoque distinto donde se prioriza el despliegue de información destacada elegida por el administrador del sitio, no un orden cronológico automático. Además, existe una serie de componentes incluidos en Liferay que facilitan la adición de recuadros con información particular a una plantilla (ROSAS, 2016).

Un problema típico en proyectos de software es que se prioriza funcionalidad sobre usabilidad de las aplicaciones, por lo cual se termina construyendo sistemas que sólo personal con un nivel intermedio o avanzado de computación puede usar. En Liferay resulta bastante intuitiva la edición de contenido en sitios web. Se puede cambiar la información general de una página navegando a través del sitio web y seleccionando “Administrar – Página”, o inclusive alguno de los bloques de una página con sólo elegir el botón Editar (ROSAS, 2016).

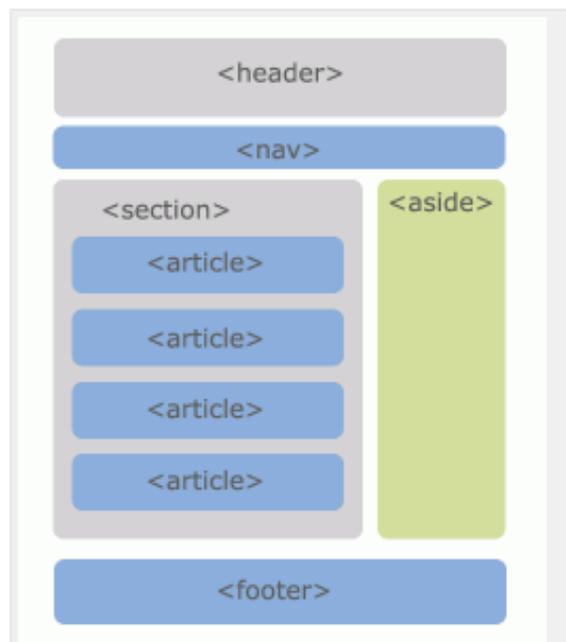


Figura 1.5 Estructura de una típica web generada con un CMS: cabecera, enlaces, artículos ordenados cronológicamente, menús laterales y pie de página (ROSAS, 2016).

Cuando un redactor quiere ingresar contenido pero que la publicación dependa de una revisión, en un gestor de contenidos sencillo no existe forma automatizada de manejar esta situación. La solución más sencilla es pedirles a los usuarios que dejen los textos en modo borrador o pendiente de revisión. Liferay cuenta con una solución completa: el motor de flujo de trabajo (*CMS Workflow*) que, una vez configurado, es capaz de manejar la situación descrita al inicio, o inclusive añadir varios pasos de revisión y edición, no sólo uno (ROSAS, 2016).

Los gestores de contenido dependen de contar con una plantilla definida, la cual sólo permite usar dos tipos de estructuras para los contenidos (una para la diagramación de la página principal y otra para las páginas internas). Al momento de crear una página en Liferay no sólo se puede elegir la apariencia de una página sino también la forma en que se distribuirá contenido en ella (ROSAS, 2016).

Una de las principales frustraciones al involucrarse en un proyecto de desarrollo de un sitio web es que el cliente parte de la premisa que es fácil incorporar pedidos adicionales, muchos de los cuales exceden las capacidades del software desarrollado o personalizado, y que requiere un mayor tiempo de desarrollo, que excede al planeamiento inicial. En Liferay se dispone de varias opciones que permitirían incorporar esos requerimientos adicionales de manera más sencilla para los desarrolladores (ROSAS, 2016):

- *Liferay Collaboration Suite.*
- *Liferay Portal Community Plugins.*
- *Desarrollo de Portlets.*

Por todo lo expresado anteriormente el Liferay fue seleccionado como plataforma corporativa, además de que el Centro de Consultoría y Desarrollo de Arquitecturas Empresariales (CDAE) posee vasta experiencia en el desarrollo de proyectos productivos, contiene funcionalidades incorporadas de gran utilidad para el sistema a implementar, las tecnologías que utiliza internamente coinciden con las utilizadas en el centro como son *Java*, *Spring*, *Hibernate* y *ServerWeb*. Este a través de wikis y foros que tiene incorporados permite fomentar el desarrollo de la gestión del conocimiento. Entre las soluciones que ofrece se encuentran:

- Portales de autoservicio.
- Espacios de trabajo para el intercambio de conocimiento.
- Sitios Web 2.0 dinámicos.
- Integración con aplicaciones corporativas.

El mismo es compatible con sus servidores de aplicaciones, bases de datos y sistemas operativos actuales, eliminando la necesidad de invertir en nuevas infraestructuras, tiene un fuerte compromiso con el cumplimiento de estándares abiertos. La adopción de estándares en las tecnologías empleadas para desarrollo de aplicaciones (*portlets*), almacenamiento y tratamiento de contenidos, servicios web y capa de presentación permite reducir el coste del desarrollo de portales basados en Liferay, que es en este caso la principal tarea (LIFERAY, 2016).

Ventajas de su uso:

- Facilita el diseño de interfaces de usuarios.
- *Framework* de integración de aplicaciones.
- Herramientas incluidas en el producto.
- Soportes de campos personalizados.
- *Plataforma Service Oriented Architecture (SOA)*.
- Configuraciones a base de *clicks*.

### **Fundamentos del Liferay**

El Liferay en cuanto a funcionalidades puede agruparse en dos conjuntos principales, uno para las funcionalidades más básicas entre las que están los roles, grupos de usuarios, etc. Estas funcionalidades hacen posible la personalización del portal de acuerdo al usuario. Por otro lado, se encuentra Liferay *Collaboration Suite*, un juego de aplicaciones que pueden utilizarse para construir comunidades de usuarios que utilicen intensivamente el portal.

Liferay IDE está basado en Eclipse, es una plataforma de programación, desarrollo y compilación de aplicaciones *Java*. Posee una atractiva interfaz que lo hace fácil y agradable de utilizar. Eclipse está construido enteramente en *Java*, pero sus alcances exceden esta instalación. Es una plataforma de desarrollo, diseñada para ser extendida de forma indefinida a través de *plugins*. En Eclipse el concepto de trabajo está basado en las perspectivas, que no es otra cosa que una preconfiguración de ventanas y editores, relacionadas entre sí, y que nos permiten trabajar en un determinado entorno de trabajo de forma óptima. Soporta el *plugins* Liferay IDE, este *plugins*

permite el desarrollo de temas, *portlets*, *hooks* y *plugins* para los portales creados con Liferay (LIFERAY, 2016).

### 1.3.3 Herramientas del Liferay

En comparación con versiones anteriores, la versión 6 de Liferay ha seguido avanzando con la disponibilidad de herramientas de desarrollo y marcos que facilitan la implementación de soluciones de forma rápida y sencilla. Liferay puede considerarse un verdadero *'framework de desarrollo'* real y completo con todo lo necesario para crear aplicaciones web complejas. Entre las herramientas más importantes que ofrece la plataforma de Liferay se encuentran las siguientes:

El **service builder** es una herramienta para Liferay, presente en el IDE, que nos permite desarrollar rápidamente aplicaciones basadas en datos. En la práctica, a partir de la definición de una entidad en formato xml se puede generar automáticamente todas las clases en la gestión de *back-end* de la persistencia de una aplicación. Todo esto se basa en el uso de Spring e Hibernate, pero la ventaja es que la herramienta realiza para nosotros la generación de distintos archivos de configuración, las clases DAO y servicios que de otro modo se haría a mano. Todo esto, por supuesto, es fácilmente extensible y personalizable (SOURCE, 2013).

**PluginSDK** es la herramienta básica para la creación de *plugins* para Liferay. Con este *plugin* podemos desarrollar *portlets*, temas, *layouts* y *hooks*.

El **Liferay IDE** es una extensión de Eclipse que nos permite crear *plugins* en un entorno integrado para Liferay. El Liferay IDE de Eclipse permite el desarrollo de *portlets*, *layouts*, temas, *hooks*, y *plug-in* en modalidad EXT. Es importante tener en cuenta que el desarrollador tiene ahora una herramienta de desarrollo especial utilizable en Eclipse, uno de los IDE más popular para el desarrollo de Java (SCARMACIO, 2010).

Para la autenticación y control de acceso incluyen mecanismos para la identificación de usuarios mediante *login* y *password* y permiten restringir el acceso a determinadas páginas a determinados usuarios. En el desarrollo de la aplicación se necesitará el uso de varios de ellos para lograr hacer un producto con la mayor calidad posible, entre ellos están:

- Spring.
- Hibernate.
- JSF(Primefaces).

**Spring 3.0.7:** Spring basándose en ficheros xml o anotaciones es el encargado de construir todos los objetos que la aplicación va a utilizar, de esta manera al ser el encargado de inicializar todos los objetos de distintos *Frameworks* es también el responsable de que se integren de forma correcta.

**Hibernate 3.6:** es una herramienta de mapeo objeto-relacional para la plataforma Java (y disponible también para .Net con el nombre de NHibernate) que facilita el mapeo de atributos entre una base de datos relacional tradicional y el modelo de objetos de una aplicación, mediante archivos declarativos (XML) que permiten establecer estas relaciones.

Como todas las herramientas de su tipo, Hibernate busca solucionar el problema de la diferencia entre los dos modelos de datos coexistentes en una aplicación: el usado en la memoria de la computadora (orientación a objetos) y el usado en las bases de datos (modelo relacional). Hibernate genera las sentencias SQL y libera al desarrollador del manejo manual de los datos que resultan de la ejecución de dichas sentencias, manteniendo la portabilidad entre todos los motores de bases de datos con un ligero incremento en el tiempo de ejecución (HIBERNATE, 2016).

**JSF 2.2.5:** soporta diversas tecnologías de presentación y ha demostrado ser poderoso en esta capa. Las principales características de JSF son:

- La interfaz del usuario es tratada como un conjunto de componentes de interfaz gráfica.
- Incluye un conjunto de APIs que le permiten representar los componentes de la interfaz de usuario y administrar su estado, maneja eventos, validar entradas, definir un esquema de navegación de las páginas y dar soporte para internacionalización y accesibilidad.

En su ciclo de vida tiene seis fases, entre las que están:

- Restaurar Vista.
- Aplicar valores de petición.
- Procesar validaciones.
- Actualizar modelo.
- Invocar aplicación.
- Producir respuesta.

JSF cuenta con muchos *frameworks* que facilitan el desarrollo de aplicaciones utilizando esta tecnología. Entre los principales *frameworks* se encuentran: Primefaces, Icefaces, Richfaces, ASFfaces y MyFaces Trinidad.

**Primefaces 5.3:** Primefaces es una librería de componentes visuales *open source* para JavaServerFaces que cuenta con un conjunto de componentes ricos que facilitan la creación de las aplicaciones web, desarrollada y mantenida por Prime Technology, una compañía turca especializada en consultoría ágil, en este caso JSF.

**Maven 3.1.0:** Maven es una herramienta de gestión de proyectos de software y comprensión. Basado en el concepto de un modelo de objeto de proyecto (POM), Maven puede gestionar acumulación, generación de informes y documentación de un proyecto desde un lugar central. Capaz de ocuparse de la gestión de un proyecto completo, desde la etapa en la que se comprueba que el código es el correcto, hasta que se despliega la aplicación.

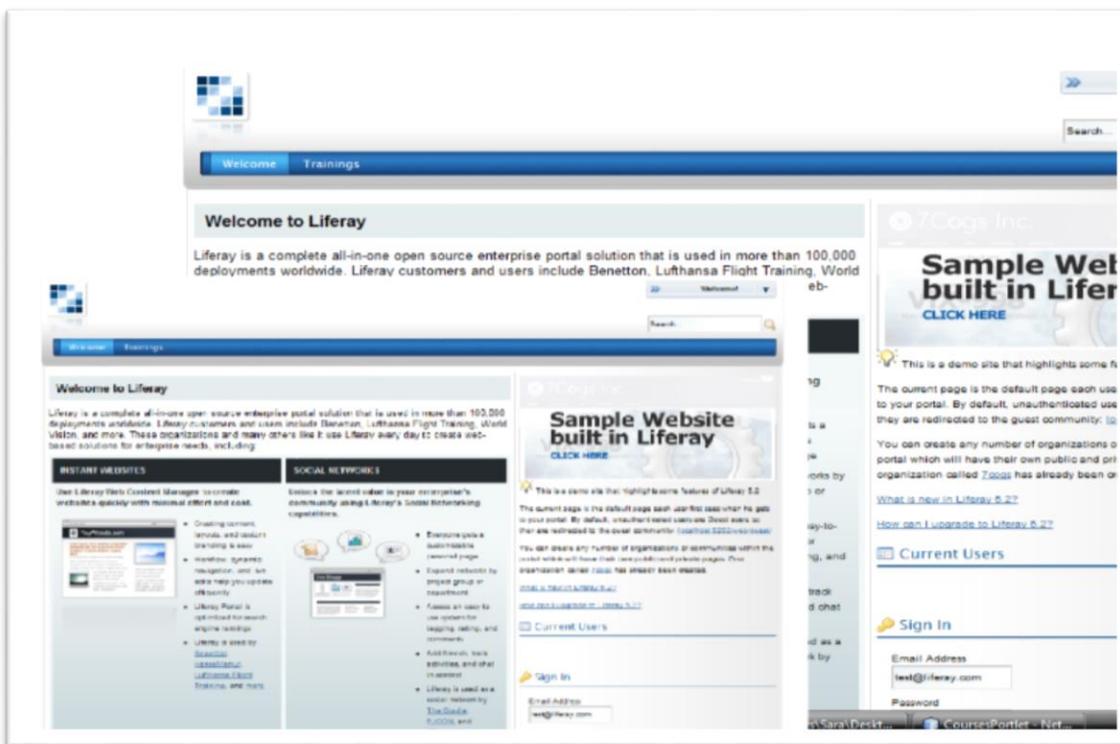


Figura 1.7 Página principal del Liferay (LIFERAY, 2016).

Esta herramienta maneja 3 niveles de repositorios, local, empresarial y global, cuando comienza el proceso de compilación, Maven busca las dependencias del proyecto en el repositorio local, si no las encuentra, intenta encontrarlas en los repositorios empresariales definidos en el *settings.xml* (archivo), y en caso de no encontrar los jar necesarios, los busca en los repositorios globales. Una vez ubicado el jar en los repositorios externos, Maven procede a descargarlo hacia su repositorio local para su correcto funcionamiento (FOUNDATION, THE APACHE SOFTWARE, 2016b).

### Uso en la Universidad de Ciencias Informáticas (UCI)

Actualmente en la universidad el Liferay junto con el trabajo de los *portlets* es un tema de estudio y no existen muchos proyectos que lo utilicen, entre estos pocos se encuentran Gestión de Operaciones del Centro de Inmunología Molecular (CIM), Portal para el Ministerio del Poder Popular para Relaciones Interiores, Justicia y Paz de Venezuela y Ministerio del Comercio Exterior y la Inversión Extranjera (MINCEX). Hoy se está persiguiendo promover este tipo de soluciones por las disimiles ventajas que trae consigo.

### 1.4. Metodologías

Dentro del desarrollo de software y a la activa necesidad de que los proyectos lleguen al éxito y obtener un producto de gran valor para nuestros clientes, generan grandes cambios en las metodologías adoptadas por los equipos para cumplir sus objetivos, puesto que, unas se adaptan

mejor que otras, al contexto del proyecto brindando mejores ventajas. Las metodologías se pueden dividir en dos grandes grupos, las tradicionales y las ágiles.

Las metodologías tradicionales se centran en la definición detallada de los procesos y tareas a realizar, herramientas a utilizar, y requiere una extensa documentación, ya que pretende prever todo de antemano. Estas metodologías no se adaptan adecuadamente a los cambios, por lo que no son métodos adecuados cuando se trabaja en un entorno donde los requisitos no pueden predecirse o pueden variar.

Por otro lado, las metodologías ágiles surgen como una mejora de las metodologías tradicionales, conteniendo nuevas características propiciando el desarrollo iterativo que permite cambios en los requisitos y posibilita que el equipo de trabajo pueda tomar decisiones a lo largo del ciclo de vida del proyecto y aplicarlas inmediatamente.

Para el desarrollo del portal se seleccionó la metodología *Agile Unified Process*, en español Proceso Unificado Ágil (AUP), la misma usa técnicas ágiles y se describe de una manera simple y fácil de entender. Entre los roles de dicha metodología se encuentran el diseñador de bases de datos, modelador, manejador de la configuración, encargado de despliegue, desarrollador, etc. Entre sus características se encuentran:

- El personal sabe lo que está haciendo: no se tiene documentación estricta, pero si hay guías y se hacen entrenamientos que permite estar al tanto del proyecto.
- Sencillez: todo se describe de manera concisa en unas cuantas páginas y no en miles de ellas.
- Agilidad: se siguen los valores y principios del manifiesto ágil y la alianza ágil.
- Enfoque en las actividades de alto valor: se enfoca en las actividades que cuentan y no en las posibles cosas que puedan suceder en el proyecto.
- Independencia de las herramientas de trabajo: no importa que herramientas se usen para hacer el trabajo.
- Se puede adaptar el producto para satisfacer necesidades individuales: no necesita ninguna herramienta especializada para adaptar AUP (FLORES, 2016).

## Conclusiones del Capítulo

Luego de realizar un estudio detallado de las principales herramientas, tecnologías y metodologías más utilizadas en el desarrollo de sistemas web, se hizo una selección de aquellas que se adaptaban más a las necesidades del cliente y que a su vez facilitaban el trabajo del equipo.

- Se citaron los conceptos fundamentales que servirán de apoyo en el desarrollo del sistema para la gestión de la actividad científica del MES.
- La metodología a utilizar.
- Se seleccionó como contenedor de *servlets* el Apache Tomcat 7.0.
- Como plataforma corporativa se acogió el Liferay 6.2 con el IDE *Eclipse Mars* 4.5.0 y lenguaje *java* 7.6.
- Se escogieron los *frameworks* Spring 3.0.7, y JSF 2.2.5 con la librería Primefaces 5.3.

- Como Objeto de Mapeo Relacional (ORM) Hibernate 3.6.
- Como sistema gestor de bases de datos se seleccionó *PostgreSQL* 9.2.4.

## CAPÍTULO 2 Planificación, diseño y desarrollo del sistema

Durante el desarrollo de este capítulo se define la arquitectura que se utilizó en el desarrollo del portal, además se muestra el modelo de datos expresando los principales conceptos y sus relaciones, se abordó algunos aspectos de la arquitectura de la información que se especifica para el portal.

### 2.1 Características del portal

El portal se caracteriza por permitir que los miembros de la entidad puedan interactuar con la información, además de potenciar la participación y el intercambio entre cada uno de los interesados promoviendo el trabajo científico-investigativo. Para lograr esto se ofrecen una serie de utilidades o aplicaciones, siendo los administradores los encargados de realizar cualquier tipo de modificación de acuerdo a las necesidades que vayan surgiendo.

A partir de una entrevista en profundidad aplicada a la directora de ciencia y técnica del MES, se detectaron como necesidades para facilitar la interacción entre los interesados y el trabajo con los proyectos de investigación científica las siguientes aplicaciones.

- Wiki
- Foros
- Blogs
- Calendario

Las mismas solucionan insuficiencias que existen en el MES y logran de manera efectiva que todos los interesados puedan intercambiar información y conocimiento de manera *online*, y con el uso del calendario pueden tener acceso de manera actualizada a cada una de las tareas que el directivo programe.

Luego de la entrevista efectuada con la directora de ciencia y técnica de MES la misma entregó un documento en el cual se reflejan los requisitos funcionales (RF) que se tuvieron en cuenta para la realización de dicho sistema. Para establecer las complejidades de cada uno de los requisitos funcionales se tuvieron en cuenta la Escala Nominal de Riesgo en Desarrollo:

- **Alta:** se otorga cuando para la implementación de los RF se considera la posible existencia de errores que lleven a la inoperatividad del código.
- **Media:** se otorga cuando pueden aparecer errores en la implementación de los RF que puedan retrasar la entrega de la versión.
- **Baja:** se otorga cuando pueden aparecer errores que serán tratados con relativa facilidad sin que traigan perjuicios para el desarrollo del proyecto.

#### Requisitos funcionales obtenidos:

No	Nombre	Descripción	Complejidad
[RF1.]	Permitir el acceso a todo profesor o investigador director de proyecto.	El sistema debe permitir que todo <b>profesor o investigador</b> director de proyecto de las universidades y ECTI adscriptas al MES puedan acceder al sistema para la introducción de datos.	Alta
[RF2.]	Permitir al usuario registrarse de forma automática.	El sistema debe dar la posibilidad de que la persona se inscriba de forma automática mediante la introducción de un nombre de usuario y una clave.	Baja
[RF3.]	Realizar la gestión de información asociada a proyectos de investigación.	El sistema debe permitir crear, modificar o eliminar un proyecto de acuerdo a los permisos creados.	Alta
[RF3.1.]	Crear nuevo proyecto.	El sistema debe tener una opción "Crear un nuevo proyecto" donde el dirigente del proyecto introducirá la información sobre el proyecto según formato que se anexa (anexo No 1). La información de esta opción se podrá guardar siempre que estén llenas todas las solicitudes.	Alta
[RF3.2.]	Modificar proyecto.	El sistema debe permitir al usuario modificar los campos de un proyecto que el mismo haya creado.	Alta
[RF3.3.]	Eliminar proyecto.	El sistema debe permitir al usuario eliminar su proyecto de investigación si este lo desea.	Alta
[RF4.]	Mostrar monto de financiamiento.	El sistema debe mostrar el monto de financiamiento y recursos por proyectos total del MES.	Alta

[RF5.]	Mostrar cantidad total de profesores y alumnos vinculados a proyectos.	El sistema debe permitir mostrar cantidad total de profesores y alumnos vinculados a proyectos.	Baja
[RF6.]	Mostrar cantidad de proyectos .	El sistema debe mostrar cantidad de proyectos PAP, PNAP, Empresariales e Institucionales.	Baja
[RF7.]	Permitir la descarga de documentos rectores.	Permitir publicación y descarga de documentos rectores de la actividad científica del MES.	Baja
[RF8.]	Gestionar roles y permisos.	Permitir la gestión de roles y permisos a nivel de <i>portlet</i> .	Baja
[RF9.]	Listar proyectos.	El sistema debe listar proyectos.	Alta
[RF10.]	Realizar aval científico.	El sistema debe permitir que el personal calificado evalúe el proyecto de investigación y en dependencia de esto lo avale o no.	Alta
[RF11.]	Realizar comentarios a través de blogs.	El sistema debe permitir a los usuarios realizar un blog a modo de diario personal donde agreguen contenidos de su interés actualizándose con frecuencia y que pueda ser comentado por sus lectores.	Media
[RF12.]	Contener wikis.	El sistema debe permitir contener wikis donde las mismas puedan ser editada directamente desde el navegador donde los usuarios crean, modifican o eliminan contenidos que, generalmente, comparten.	Media
[RF13.]	Permitir a usuarios compartir.	El sistema debe permitir a través de foros el intercambio entre personas que deseen discutir sobre problemáticas específicas.	Media

	conocimientos a través de foros.		
[RF14.]	Crear tema visual según pautas de diseño.	El sistema debe contener un tema visual que se adapte a las características del MES.	Alta
[RF15.]	Visualizar proyectos avalados.	El sistema debe mostrar los proyectos que han sido avalados por el personal calificado.	Alta
[RF16.]	Realizar reportes.	El sistema debe permitir realizar reportes, mostrando los proyectos creados dado una fecha o todos los proyectos en general.	Alta
[RF17.]	Contener una autenticación única.	El sistema debe permitir que el portal se autentique a través de un mecanismo centralizado SSO.	Alta
[RF18.]	Integrar el sistema con el LDAP del MES.	El sistema debe importar credenciales de un <i>Lightweight Directory Access Protocol</i> (LDAP).	Alta
[RF19.]	Contener un calendario.	Permitir la visualización de las principales actividades del MES.	Media
[RF20.]	Proponer arquitectura para despliegue.	El sistema debe contener una propuesta de despliegue en un escenario de alta disponibilidad que incluya balanceo de carga.	Alta
[RF21.]	Contener gráficas representativas.	El sistema debe mostrar gráficas por tipo de proyectos, cantidad de personal por proyecto y una para mostrar el financiamiento en cada moneda por tipo de proyecto.	Media

Tabla 2.1 Requisitos funcionales.

Entre los requisitos no funcionales (RNF) que no son más que las propiedades o cualidades que el sistema debe tener y representan las características que hacen el producto más atractivo, usable, rápido o confiable para los usuarios se encuentran:

No	Nombre	Descripción
[RNF1.]	Software.	El sistema debe utilizar la máquina virtual en su versión 7.
[RNF2.]	Software.	El sistema deberá ser multiplataforma.
[RNF3.]	Hardware.	La capacidad requerida de memoria RAM será igual o mayor a 6 Gigabyte.
[RNF4.]	Hardware.	El disco duro deberá tener como mínimo 80 Gigabyte.
[RNF5.]	Diseño.	El sistema debe contar con un diseño adaptables a las resoluciones de diferentes dispositivos.

Tabla 2.2 Requisitos no funcionales.

## 2.2 Modelo de dominio

Para lograr un mejor entendimiento entre clientes, desarrolladores e interesados se propone un modelo de dominio, donde de manera visual permite mostrar los principales conceptos que se manejan en el dominio del sistema en desarrollo.

Lo conceptos se relacionan de la siguiente forma: el portal contiene proyectos de investigación, una funcionalidad llamada expertos, wikis, foros y blogs, estos últimos serán creados por el usuario, permitiéndole a su vez insertar los proyectos. Los proyectos tendrán una institución y un área a la que pertenecen, tienen una línea de investigación, un tipo, un impacto que puede ser positivo o negativo y tareas, que, a su vez, estas podrán contener subtareas.

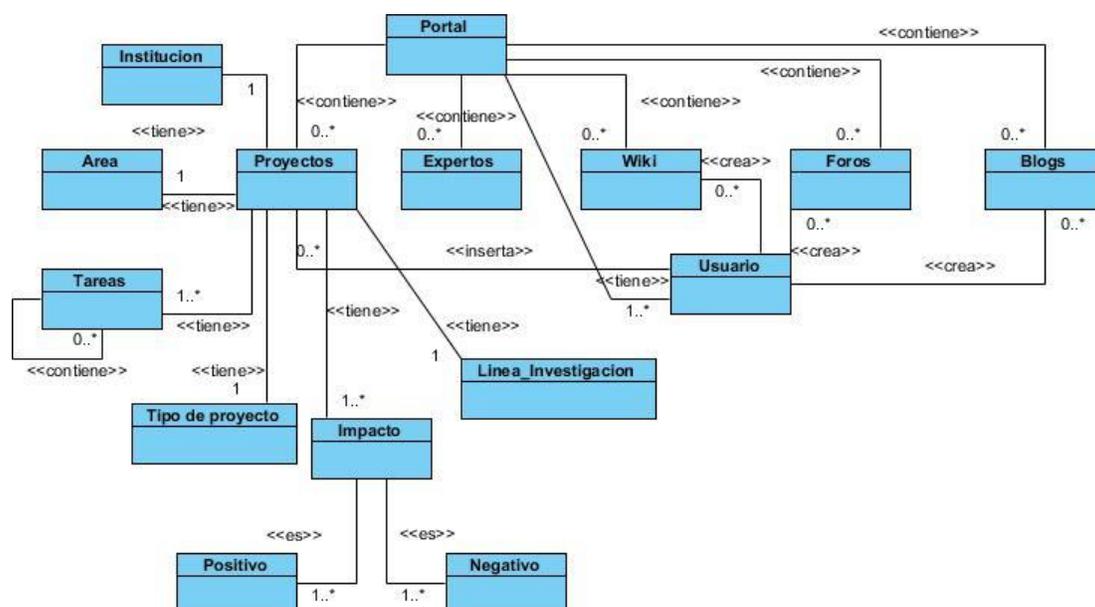


Figura 2.1 Modelo de Dominio

### 2.3 Arquitectura del portal

La Arquitectura Orientada a Servicio es la arquitectura por referencia del Liferay, este posee una arquitectura basada en un sistema jerárquico y extensible de comunidades y organizaciones, las cuales son replicadas mediante plantillas. Las comunidades son un entorno de trabajo en la que todos los usuarios comparten conocimientos de un tema común logrando con esto darle mantenimiento a la misma. La arquitectura del portal está ideada en términos de presentación, lógica del negocio y el acceso a datos, donde cada una de ellas está estrechamente vinculada con las demás.

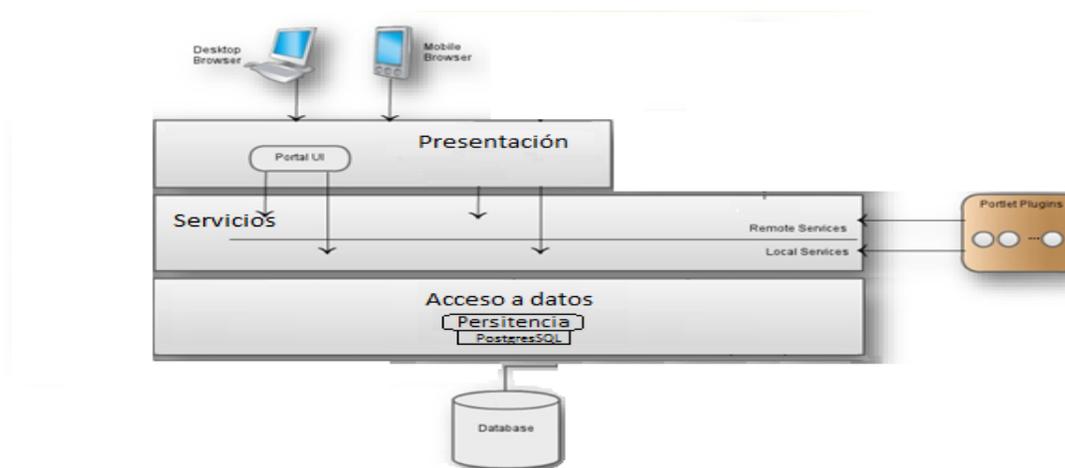


Figura 2.2. Arquitectura del portal (LIFERAY, 2016).

La capa de servicios se divide en dos subcapas: servicios locales (<Entity> LocalService.java) y servicios remotos (<Entity> Service.java). La misma contiene la gran mayoría de la lógica de negocio para la plataforma de portal y todos los *portlets* incluido fuera de la caja. Los servicios están organizados en dos grupos: *Portal Services* y *Portlet Services*. El primero contiene los servicios para los niveles de las entidades del portal y el segundo contiene los servicios utilizados por los diferentes *portlet* del portal(LIFERAY, 2016).

*Local Services:* este contiene la lógica del negocio y comunica con la capa persistente. *Remote Services* tiene como principal objetivo realizar comprobaciones de seguridad antes de invocar el método equivalente en el servicio local. *Service Builder* es la herramienta que mantiene unida todas las capas del Liferay y que oculta la complejidad de la utilización de la Spring o Hibernate(LIFERAY, 2016).

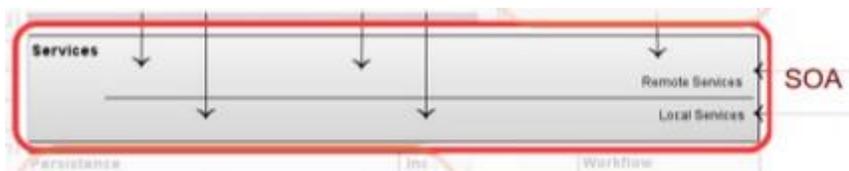


Figura 2.3. Capa de servicios (LIFERAY, 2016).

### 2.3.1 Estilos y patrones arquitectónico utilizados

El estilo arquitectónico en capas, en este caso, en tres capas fue el utilizado, su objetivo primordial la separación de la capa de presentación, capa de negocio y la capa de acceso a datos. El desarrollo se puede llevar a cabo en varios niveles y en caso de sobrevenga algún cambio es mucho más fácil la manipulación sin interferir en las demás capas. Fue seleccionado ya que para el trabajo en la web trae muchas facilidades como es que los datos y servicios están separados y es fácil separar los datos de “la lógica del negocio”.

Por otra parte, el cliente recibe los datos y la información de forma indirecta a través del servidor. Se fomenta la escalabilidad del sistema, la seguridad y la reutilización de código. Los principales beneficios de un estilo arquitectónico en capas son la abstracción, aislamiento, rendimiento, mejoras en pruebas, y la independencia que permite. Miembros de la familia son las arquitecturas llamada y retorno, el modelo-vista-controlador, los sistemas orientados a objeto y los sistemas basados en componentes (PELÁEZ, 2009).

#### Patrón Modelo Vista Controlador.

Modelo Vista Controlador (MVC): es un patrón de arquitectura de las aplicaciones software, separa la lógica de negocio de la interfaz de usuario, este facilita la evolución por separado de ambos aspectos e incrementa la reutilización y flexibilidad. Para las aplicaciones web en la vista encontramos la página xhtml, en el Contralor se encuentra el código que obtiene datos dinámicos y genera el contenido xhtml, y en el modelo se encuentra la información almacenada en la base de datos, junto con las reglas de negocio que transforman esa información (PAVÓN, 2008).

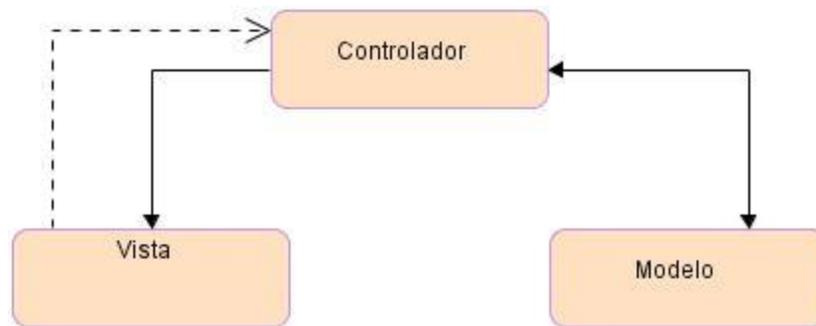


Figura 2.4. Patrón de diseño Modelo Vista Controlador.

En el caso de JSF se ve ejemplificado el patrón MVC, donde el mismo corresponde a la siguiente estructura:

- Modelo: en el modelo las clases que lo componen son proyecto, director, tarea e impacto.
- Vista: en la vista se encuentra añadir, mostrar y modificar.
- Controlador: en el controlador estará la clase Crud\_Manager\_Bean.

### 2.3.2 Patrones de diseño

Los patrones de diseño son soluciones para problemas típicos y recurrentes que nos podemos encontrar la hora de desarrollar una aplicación. El uso de los mismos nos permite crear

aplicaciones robustas y fáciles de mantener. “Los patrones de diseño son el esqueleto de las soluciones a problemas comunes en el desarrollo de software.” En otras palabras, brindan una solución ya probada y documentada a problemas de desarrollo de software que están sujetos a contextos similares (TEDESCHI, 2016). Existen varios patrones de diseño los cuales se clasifican de la siguiente manera:

- Patrones Creacionales: inicialización y configuración de objetos.
- Patrones Estructurales: separan la interfaz de la implementación. Se ocupan de cómo las clases y objetos se agrupan, para formar estructuras más grandes.
- Patrones de Comportamiento: más que describir objetos o clases, describen la comunicación entre ellos.

Entre los patrones utilizados para el diseño del portal para la gestión de la información asociada a los proyectos científicos del MES se encuentran:

- *Composite view*: este patrón ayuda al proceso de integración de varias subvistas en una página. Normalmente una funcionalidad está compuesta por un conjunto de vistas asociadas. Es habitual que varias subvistas se integren para completar una sola página. Además, varios individuos con diferentes habilidades contribuyen al desarrollo y mantenimiento de esas páginas web (ORACLE, 2016). En el caso del portal para la gestión de la información asociada a la actividad científica del MES las subvistas serán los *portlets* implementados como el *listarProyecto* y *gestionarProyecto*.

### **Modelo de datos.**

El modelo de entidad relación es un modelo de datos basado en una percepción del mundo real que consiste en un conjunto de objetos básicos llamados entidades y relaciones entre estos objetos, implementándose en forma gráfica a través del diagrama entidad relación (GUILLERMO STORTI, 2007). En el caso del Liferay tiene un mecanismo que permite a través del diseño de clases generar las diferentes capas de presentación y se generen a su vez físicamente los datos en un diagrama de entidad relación que lo guarda en un gestor de base de datos, en este caso *postgres*.

Para la implementación del portal se hizo necesario añadir a las bases de datos Liferay las siguientes tablas referentes a un conjunto de funcionalidades:

**Proyecto:** en esta tabla se almacena toda la información de los proyectos que se insertan, como son el título, descripción, objetivos generales, objetivos específicos, resultados esperados, la fecha en que comienza y termina cada proyecto de investigación, profesores y estudiantes vinculados, quien lo elabora, si está avalado o no y la institución a la que pertenece. Esta tabla tiene relación con todas las tablas existentes.

**Institución:** en esta tabla se almacena el nombre de la institución a la que corresponde el proyecto que se inserte y posee una relación por tanto con la tabla proyecto.

**Usuario:** en esta tabla se guardan los datos del usuario como nombre, cargo y email, tiene una relación con la tabla proyecto.

Linea\_Invest: en esta tabla se almacena la línea de investigación que sigue un determinado proyecto por lo que está relacionada con dicha tabla.

Tipo\_Proyecto: en esta tabla se guardan el nombre del tipo de proyecto al que puede pertenecer por lo que está relacionada con la tabla proyecto.

Área: en esta tabla se almacena el área a la que pertenece un proyecto determinado por lo que se relaciona directamente con la tabla proyecto.

Impacto: en esta tabla se almacenan los impactos que puede traer consigo un proyecto dado, relacionándose con la tabla proyecto.

Tarea: en esta tabla se almacenan las diferentes tareas que encierra un proyecto y a su vez contienen las subtareas que puede poseer una tarea, relacionándose con ella misma y con la tabla proyecto.

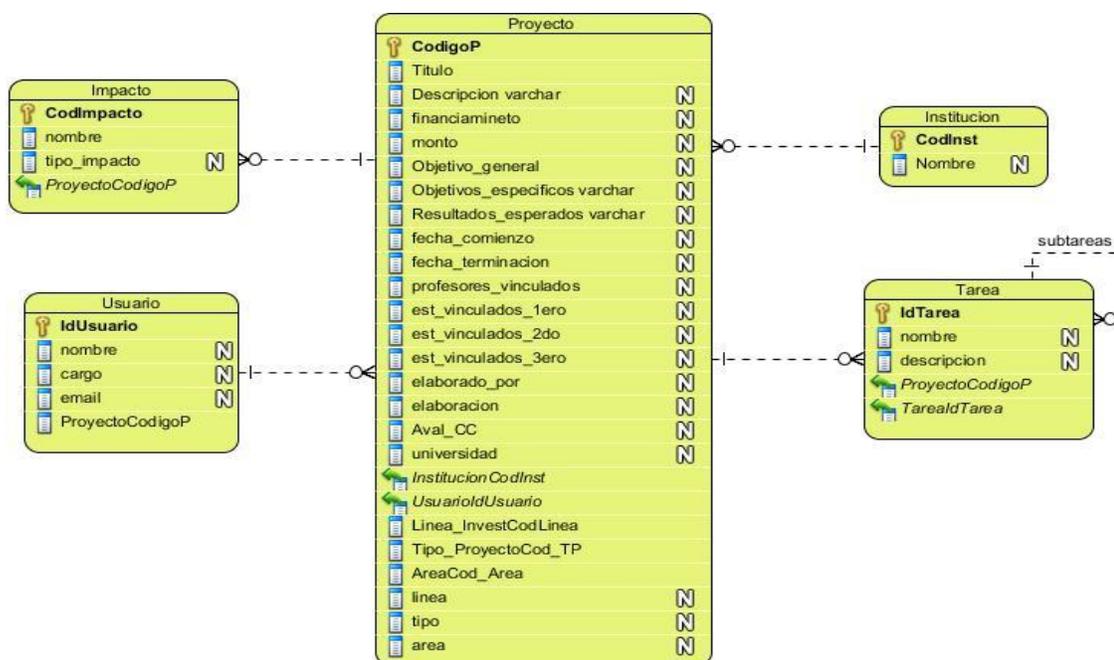


Figura 2.5. Modelo conceptual de datos de la gestión de proyectos.

A continuación, se muestra en la figura 2.6 el modelo entidad relación que se diseña con la herramienta service builder, donde en una vista más detallada se pueden ver los datos.

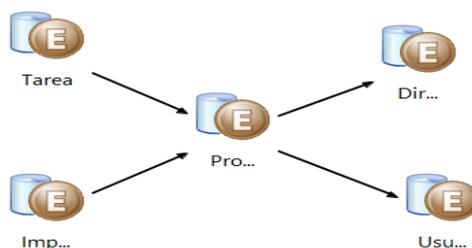


Figura 2.6 service.xml.

## 2.4 Historias de usuario

Las historias de usuario (HU) son tarjetas en donde el interesado describe brevemente las características que el sistema debe poseer, sean requisitos funcionales o no funcionales. Son utilizados en metodologías ágiles, como es nuestro caso, donde cada historia de usuario debe ser lo suficientemente comprensible y delimitada para que se pueda implementar de una manera rápida. Se utilizó la herramienta Visual Paradigm para la creación de los prototipos realizando validaciones a través de los mismo.

### 2.4.1 Descripción de las HU

#### HU #2 Registrar usuario.

Historia de Usuario	
<b>Número:</b> 1.	<b>Usuario:</b> administrador, avalador, usuario.
<b>Modificación de Historia de Usuario #:</b> ninguna	
<b>Nombre de Historia de Usuario:</b> registrar usuario en el sistema.	
<b>Prioridad en negocio:</b> muy alta.	<b>Riesgo de Desarrollo:</b> alta.
<b>Puntos estimados:</b> 1.0	<b>Iteración asignada:</b> 2.
<b>Programador(es) responsable(s):</b> Raúl Pérez Guevara, Yessida Tamayo Barrio.	
<b>Descripción:</b> el sistema debe permitir que el usuario se registre a través de un usuario, cargo, y el email correspondiente.	
<b>Observaciones:</b>	
<b>Prototipo de interfaz:</b>	
	

Tabla 2.3 Historia de usuario: registrar usuario.

HU #3.1 Insertar Proyecto.

Historia de Usuario																									
Número: 2.	Usuario: administrador, avalador, usuario.																								
Modificación de Historia de Usuario #: ninguna																									
Nombre de Historia de Usuario: crear nuevo proyecto.																									
Prioridad en negocio: muy alta.	Riesgo de Desarrollo: alta.																								
Puntos estimados: 1.0	Iteración asignada: 4.																								
Programador(es) responsable(s): Raúl Pérez Guevara.																									
<p><b>Descripción:</b> el sistema debe tener una opción “crear un nuevo proyecto” donde el dirigente del proyecto introducirá la información sobre el proyecto según formato que se anexa (anexo No 1). La información de esta opción se podrá guardar siempre que estén llenas todas las solicitudes.</p>																									
<p><b>Observaciones:</b> el usuario debe estar registrado en el sistema.</p>																									
<p><b>Prototipo de interfaz:</b></p> <div style="border: 1px solid black; padding: 10px;"> <p>Título  <input type="text"/></p> <p>Fecha inicio <input type="text" value="22/2/2016"/> Fecha fin <input type="text" value="22/2/2017"/></p> <p>Universidad <input type="text"/> Tipo <input type="text"/> Área <input type="text"/></p> <p>Línea a la que está involucrado  <input type="text"/></p> <p>Descripción del proyecto  <input type="text" value="Dos importantes vacunas cubanas, una contra la tosferina, y otra para combatir el cólera....."/></p> <p>Objetivos(Generales y Especificos)  <input type="text"/></p> <p>Impacto esperado  <input type="text"/></p> <p>Nombre de la institución  <input type="text"/></p> <p>Nombre completo del director del proyecto  <input type="text"/></p> <p>Cargo del director del proyecto  <input type="text"/></p> <p>Involucrados (Cantidad de profesores involucrados)  <input type="text"/></p> <table border="0"> <tr> <td colspan="2">Cantidad de alumnos involucrados en el proyecto</td> <td colspan="2">Impacto Medioambiental</td> </tr> <tr> <td>1ero</td> <td><input type="text" value="12"/></td> <td><input type="checkbox"/> Positivo</td> <td><input type="checkbox"/> Negativo</td> </tr> <tr> <td>2do</td> <td><input type="text" value="12"/></td> <td><input type="checkbox"/> Mejoramiento de la capacidad productiva del suelo y/o disminución de los índices</td> <td><input type="checkbox"/> Degradamiento del suelo</td> </tr> <tr> <td>3ero</td> <td><input type="text" value="12"/></td> <td><input type="checkbox"/> Mejoría de las condiciones de accesibilidad y transporte</td> <td><input type="checkbox"/> Contaminación</td> </tr> <tr> <td>4to</td> <td><input type="text" value="12"/></td> <td><input type="checkbox"/> Incremento de la superficie boscosa</td> <td><input type="checkbox"/> Daños a la salud humana.</td> </tr> <tr> <td>5to</td> <td><input type="text" value="12"/></td> <td></td> <td></td> </tr> </table> <p style="text-align: right;"><input type="button" value="Insertar"/></p> </div>		Cantidad de alumnos involucrados en el proyecto		Impacto Medioambiental		1ero	<input type="text" value="12"/>	<input type="checkbox"/> Positivo	<input type="checkbox"/> Negativo	2do	<input type="text" value="12"/>	<input type="checkbox"/> Mejoramiento de la capacidad productiva del suelo y/o disminución de los índices	<input type="checkbox"/> Degradamiento del suelo	3ero	<input type="text" value="12"/>	<input type="checkbox"/> Mejoría de las condiciones de accesibilidad y transporte	<input type="checkbox"/> Contaminación	4to	<input type="text" value="12"/>	<input type="checkbox"/> Incremento de la superficie boscosa	<input type="checkbox"/> Daños a la salud humana.	5to	<input type="text" value="12"/>		
Cantidad de alumnos involucrados en el proyecto		Impacto Medioambiental																							
1ero	<input type="text" value="12"/>	<input type="checkbox"/> Positivo	<input type="checkbox"/> Negativo																						
2do	<input type="text" value="12"/>	<input type="checkbox"/> Mejoramiento de la capacidad productiva del suelo y/o disminución de los índices	<input type="checkbox"/> Degradamiento del suelo																						
3ero	<input type="text" value="12"/>	<input type="checkbox"/> Mejoría de las condiciones de accesibilidad y transporte	<input type="checkbox"/> Contaminación																						
4to	<input type="text" value="12"/>	<input type="checkbox"/> Incremento de la superficie boscosa	<input type="checkbox"/> Daños a la salud humana.																						
5to	<input type="text" value="12"/>																								

--

Tabla 2.4 Historia de usuario: crear nuevo proyecto.

**HU #3.2 Modificar proyecto.**

Historia de Usuario		
<b>Número:</b> 3.	<b>Usuario:</b> administrador, avalador, usuario.	
<b>Modificación de Historia de Usuario #:</b> ninguna		
<b>Nombre de Historia de Usuario:</b> modificar proyecto.		
<b>Prioridad en negocio:</b> muy alta.	<b>Riesgo de Desarrollo:</b> alta.	
<b>Puntos estimados:</b> 1.0	<b>Iteración asignada:</b> 4.	
<b>Programador(es) responsable(s):</b> Raúl Pérez Guevara, Yessida Tamayo Barrio.		
<b>Descripción:</b> el sistema debe permitir al usuario modificar los campos de un proyecto que el mismo haya creado.		
<b>Observaciones:</b> el usuario debe estar registrado en el sistema, y el id del usuario que quiere realizar los cambios debe coincidir con el id del usuario que haya creado el proyecto.		
<b>Prototipo de interfaz:</b>		
<p>Título</p> <input style="width: 500px; height: 20px;" type="text" value="xxxxxxxxxxxxxxxxxxxxxxxx"/>		
Fecha inicio	Fecha fin	
<input style="width: 80px; height: 20px;" type="text" value="22/2/2016"/>	<input style="width: 80px; height: 20px;" type="text" value="22/2/2017"/>	
Universidad	Tipo	Área
<input style="width: 100px; height: 20px;" type="text" value="v"/>	<input style="width: 100px; height: 20px;" type="text" value="v"/>	<input style="width: 100px; height: 20px;" type="text" value="v"/>
Línea a la que está involucrado		
<input style="width: 500px; height: 20px;" type="text" value="xxxxxxxxxxxx"/>		
Descripción del proyecto		
<input style="width: 500px; height: 20px;" type="text" value="Dos importantes vacunas cubanas, una contra la tosferina, y otra para combatir el cólera....."/>		
<input style="width: 60px; height: 20px;" type="button" value="Modificar"/>		

Tabla 2.5 Historia de usuario: modificar proyecto.

**HU #3.3 Eliminar proyecto.**

Historia de Usuario	
<b>Número:</b> 4	<b>Usuario:</b> administrador.
<b>Modificación de Historia de Usuario #:</b> ninguna	
<b>Nombre de Historia de Usuario:</b> eliminar proyecto.	
<b>Prioridad en negocio:</b> muy alta.	<b>Riesgo de Desarrollo:</b> alta.
<b>Puntos estimados:</b> 1.0	<b>Iteración asignada:</b> 4.
<b>Programador(es) responsable(s):</b> Raúl Pérez Guevara, Yessida Tamayo Barrio.	
<b>Descripción:</b> el sistema debe permitir al usuario eliminar el proyecto que el mismo haya creado.	
<b>Observaciones:</b> el id del usuario que esté registrado debe coincidir con el id del usuario que creo el proyecto para que el mismo pueda ser eliminado.	
<b>Prototipo de interfaz:</b>	
	

Tabla 2.6 Historia de usuario: eliminar proyecto.

**HU # 7 Permitir la descarga de documentos rectores.**

Historia de Usuario	
<b>Número:</b> 5	<b>Usuario:</b> administrador, avalador, usuario.
<b>Modificación de Historia de Usuario #:</b> ninguna	
<b>Nombre de Historia de Usuario:</b> permitir la descarga de documentos rectores.	
<b>Prioridad en negocio:</b> medio	<b>Riesgo de Desarrollo:</b> medio.

<b>Puntos estimados:</b> 1.0.	<b>Iteración asignada:</b> 1.
<b>Programador(es) responsable(s):</b> Yessida Tamayo Barrio.	
<b>Descripción:</b> permitir publicación y descarga de documentos rectores de la actividad científica del MES.	
<b>Observaciones:</b> esto podrá solo introducirse desde el MES.	
<b>Prototipo de interfaz:</b>	
	

Tabla 2.7 Historia de usuario: permitir la descarga de documentos rectores.

#### HU # 4 Mostrar monto de financiamiento.

Historia de Usuario	
<b>Número:</b> 5.	<b>Usuario:</b> administrador, avalador, usuario.
<b>Modificación de Historia de Usuario #:</b> ninguna	
<b>Nombre de Historia de Usuario:</b> mostrar monto de financiamiento.	
<b>Prioridad en negocio:</b> media.	<b>Riesgo de Desarrollo:</b> medio.
<b>Puntos estimados:</b> 0.8.	<b>Iteración asignada:</b> 2.
<b>Programador(es) responsable(s):</b> Yessida Tamayo Barrio.	
<b>Descripción:</b> el sistema debe mostrar el monto de financiamiento y recursos por proyectos total del MES.	
<b>Observaciones:</b> -	
<b>Prototipo de interfaz:</b>	



Tabla 2.8 Historia de usuario: mostrar monto de financiamiento.

**8 HU # 5 Mostrar cantidad total de profesores y alumnos vinculados a proyectos.**

Historia de Usuario	
<b>Número:</b> 5.	<b>Usuario:</b> administrador, avalador, usuario.
<b>Modificación de Historia de Usuario #:</b> ninguna	
<b>Nombre de Historia de Usuario:</b> mostrar cantidad total de profesores y alumnos vinculados a un proyecto.	
<b>Prioridad en negocio:</b> media.	<b>Riesgo de Desarrollo:</b> medio.
<b>Puntos estimados:</b> 0.8.	<b>Iteración asignada:</b> 2.
<b>Programador(es) responsable(s):</b> Yessida Tamayo Barrio.	
<b>Descripción:</b> el sistema debe permitir mostrar cantidad total de profesores y alumnos vinculados a proyectos.	
<b>Observaciones:</b> -	
<b>Prototipo de interfaz:</b>	
<p>The image shows a web interface with three dropdown menus at the top labeled 'Financiamiento', 'Personal', and 'Proyectos'. Below them is a vertical menu with the following items: 'Personal adj...', 'Obtener', 'Gráficas', and another 'Obtener'. The 'Obtener' and 'Gráficas' items are highlighted with a dark border.</p>	

Tabla 2.9 Historia de usuario: mostrar cantidad de profesores y alumnos vinculados a un proyecto.

**HU # 6 Mostrar cantidad de proyectos.**

Historia de Usuario	
<b>Número:</b> 5.	<b>Usuario:</b> administrador, avalador, usuario.
<b>Modificación de Historia de Usuario #:</b> ninguna	
<b>Nombre de Historia de Usuario:</b> mostrar cantidad de proyectos.	
<b>Prioridad en negocio:</b> media.	<b>Riesgo de Desarrollo:</b> medio.
<b>Puntos estimados:</b> 0.8.	<b>Iteración asignada:</b> 2.
<b>Programador(es) responsable(s):</b> Yessida Tamayo Barrio.	
<b>Descripción:</b> el sistema debe mostrar cantidad de proyectos PAP, PNAP, Empresariales e Institucionales.	
<b>Observaciones:</b> -	
<b>Prototipo de interfaz:</b>	
	

Tabla 2.9 Historia de usuario: mostrar cantidad de proyectos.

**HU #9 Listar Proyectos.**

Historia de Usuario	
<b>Número:</b> 6.	<b>Usuario:</b> administrador, avalador, usuario.
<b>Modificación de Historia de Usuario #:</b> ninguna	
<b>Nombre de Historia de Usuario:</b> listar resultados de la ejecución.	
<b>Prioridad en negocio:</b> muy alta.	<b>Riesgo de Desarrollo:</b> alta.
<b>Puntos estimados:</b> 0.6	<b>Iteración asignada:</b> 2.

<b>Programador(es) responsable(s):</b> Yessida Tamayo Barrio, Raúl Pérez Guevara.
<b>Descripción:</b> mostrará todos los proyectos y permitirá al investigador ver cada uno de los proyectos subidos al sistema.
<b>Observaciones:</b> el usuario debe encontrarse registrado en el sistema.
<b>Prototipo de interfaz:</b>


Tabla 2.10 Historia de usuario: listar resultados de ejecución.

**HU #10 Realizar aval científico.**

Historia de Usuario	
<b>Número:</b> 7.	<b>Usuario:</b> avalador.
<b>Modificación de Historia de Usuario #:</b> ninguna	
<b>Nombre de Historia de Usuario:</b> realizar aval científico.	
<b>Prioridad en negocio:</b> alta.	<b>Riesgo de Desarrollo:</b> medio.
<b>Puntos estimados:</b> 0.8.	<b>Iteración asignada:</b> 2.
<b>Programador(es) responsable(s):</b> Raúl Pérez Guevara, Yessida Tamayo Barrio.	
<b>Descripción:</b> en esta opción el dirigente del proyecto introducirá un documento en formato pdf que contenga el aval del Consejo Científico de la Universidad o ECTI.	
<b>Observaciones:</b> el usuario debe encontrarse registrado en el sistema	
<b>Prototipo de interfaz:</b>	

The form contains the following fields and options:

- Título:** Text input field with placeholder text.
- Fecha inicio:** Date input field (22/2/2016).
- Fecha fin:** Date input field (22/2/2017).
- Universidad:** Dropdown menu.
- Tipo:** Dropdown menu.
- Área:** Dropdown menu.
- Línea a la que está involucrado:** Text input field.
- Descripción del proyecto:** Text input field.
- Objetivos(Generales y Específicos):** Text input field.
- Impacto esperado:** Text input field.
- Nombre de la institución:** Text input field.
- Nombre completo del director del proyecto:** Text input field.
- Cargo del director del proyecto:** Text input field.
- Involucrados (Cantidad de profesores involucrados):** Text input field.
- Cantidad de alumnos involucrados en el proyecto:** Five input fields labeled 1ero, 2do, 3ero, 4to, 5to, each with a value of 12.
- Impacto Medioambiental:**
  - Positivo:**
    - Mejoramiento de la capacidad productiva del suelo y/o disminución de los índices
    - Mejoría de las condiciones de accesibilidad y transporte
    - Incremento de la superficie boscosa
  - Negativo:**
    - Degradamiento del suelo
    - Contaminación
    - Daños a la salud humana.
- Buttons:** 'Insertar' and 'Avalar'.

Tabla 2.11 Historia de usuario: realizar aval científico.

**HU #12 Contener wikis.**

Historia de Usuario	
<b>Número:</b> 8	<b>Usuario:</b> administrador, avalador, usuario.
<b>Modificación de Historia de Usuario #:</b> ninguna	
<b>Nombre de Historia de Usuario:</b> contener wikis	
<b>Prioridad en negocio:</b> media.	<b>Riesgo de Desarrollo:</b> baja
<b>Puntos estimados:</b> 1.0	<b>Iteración asignada:</b> 2.
<b>Programador(es) responsable(s):</b> Yessida Tamayo Barrio.	
<b>Descripción:</b> el sistema debe permitir contener wikis donde la misma pueda ser editada directamente desde el navegador donde los usuarios crean, modifican o eliminan contenidos que, generalmente, comparten.	
<b>Observaciones:</b> el usuario debe estar registrado en el sistema.	
<b>Prototipo de interfaz:</b>	

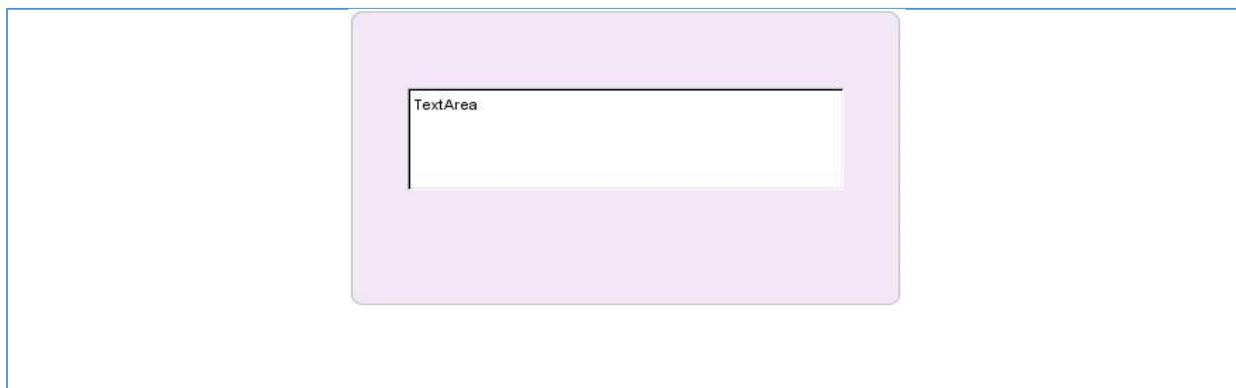


Tabla 2.12 Historia de usuario: contener wikis.

**HU #13 Realizar foros.**

Historia de Usuario	
<b>Número:</b> 9	<b>Usuario:</b> administrador, avalador, usuario.
<b>Modificación de Historia de Usuario #:</b> ninguna	
<b>Nombre de Historia de Usuario:</b> permitir a usuarios compartir conocimientos a través de foros.	
<b>Prioridad en negocio:</b> media.	<b>Riesgo de Desarrollo:</b> baja.
<b>Puntos estimados:</b> 1.0	<b>Iteración asignada:</b> 2.
<b>Programador(es) responsable(s):</b> Yessida Tamayo Barrio.	
<b>Descripción:</b>	
<b>Observaciones:</b> el usuario debe estar registrado en el sistema.	
<b>Prototipo de interfaz:</b>	

Tabla 2.13 Historia de usuario: realizar foros.

**HU #16 Realizar reportes.**

Historia de Usuario	
<b>Número:</b> 10	<b>Usuario:</b> administrador, avalador, usuario.
<b>Modificación de Historia de Usuario #:</b> ninguna	
<b>Nombre de Historia de Usuario:</b> realizar reportes.	
<b>Prioridad en negocio:</b> alta.	<b>Riesgo de Desarrollo:</b> alta.
<b>Puntos estimados:</b> 1.0	<b>Iteración asignada:</b> 2.
<b>Programador(es) responsable(s):</b> Raúl Pérez Guevara, Yessida Tamayo Barrio.	
<b>Descripción:</b> el sistema debe permitir realizar reportes, mostrando los proyectos creados dado una fecha.	
<b>Observaciones:</b> el usuario debe estar registrado en el sistema.	
<b>Prototipo de interfaz:</b>	
	

*Tabla 2.14 Historia de usuario: realizar reportes.*

**2.4.2 Modelo de diseño**

Muestra los objetos o clases en un sistema, es un modelo de objetos que se centra en cómo los requisitos funcionales y no funcionales, junto con otras restricciones relacionadas con el entorno de implementación, tienen impacto en el sistema a considerar. Es el puente en los requerimientos y la implementación del sistema.

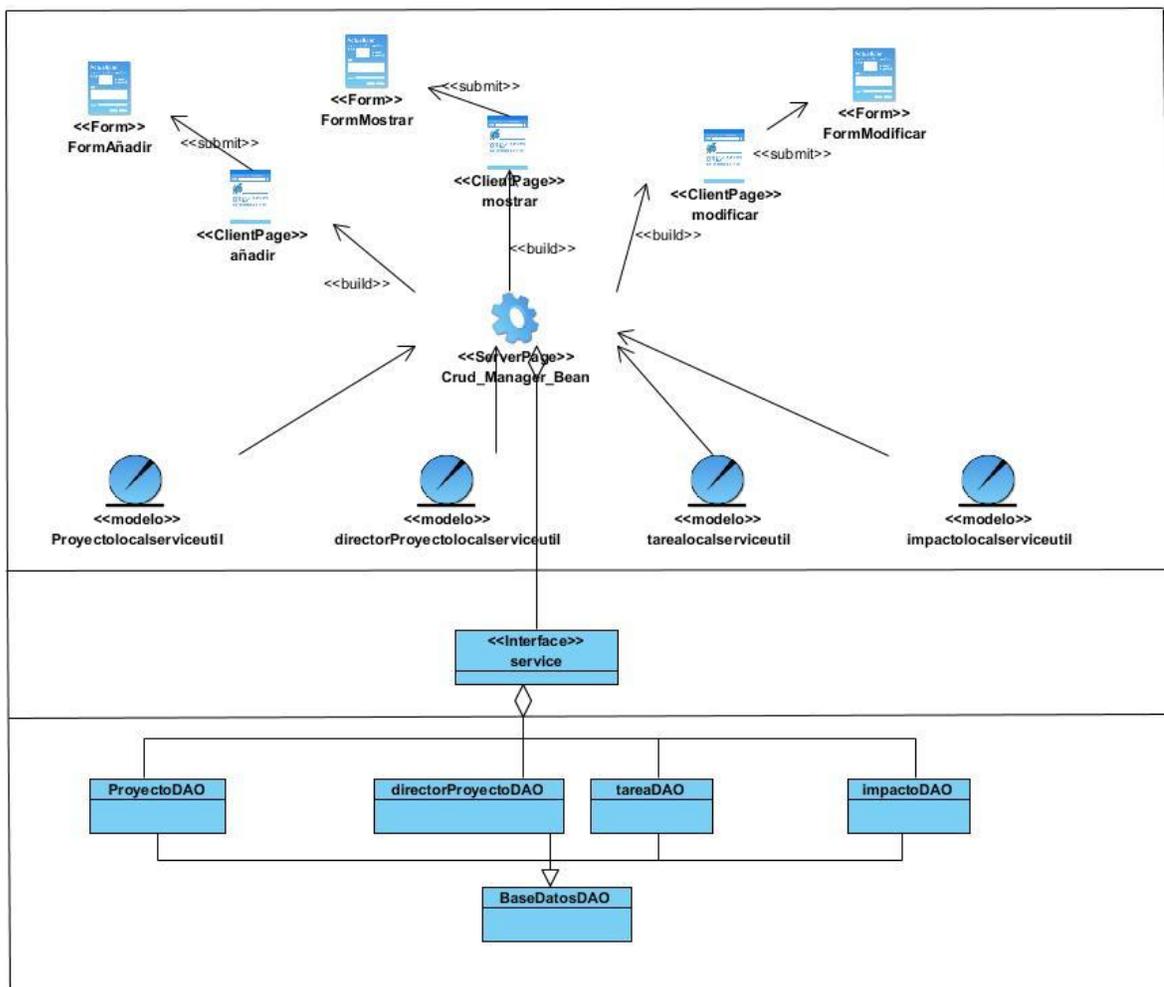


Figura 2.7. Diagrama de clases de diseño.

**Descripción de las clases de diseño.**

Nombre	Descripción	Tipo
<b>Añadir</b>	Interfaz mostrada al usuario que le brinda al usuario la posibilidad de llenar los campos requeridos.	Interfaz
<b>Mostrar</b>	Interfaz que le muestra al usuario una tabla con todos los proyectos subidos al sistema con los datos correspondientes.	Interfaz
<b>Modificar</b>	Interfaz que le muestra al usuario los proyecto, donde el mismo puede modificar los campos siempre y	Interfaz

	cuando este sea el que lo haya realizado.	
<b>Service</b>	Esta clase es una interfaz para hacer más fácil el acceso a las funcionalidades implementadas unificando las acciones más comunes, permitiendo comunicar la capa lógica de negocio con la capa de acceso a datos.	Interfaz
<b>Crud_Manager_Bean</b>	Es el controlador principal, que facilita el manejo de las peticiones realizadas por los usuarios, además, es el encargado de comunicar la capa de presentación con la capa lógica de negocio.	Controladora
<b>Proyectolocalserviceutil</b>	Esta clase almacena toda la información sobre los proyectos.	Modelo
<b>DirectorProyectolocalserviceutil</b>	Esta clase almacena toda la información acerca del director del proyecto.	Modelo
<b>tarealocalserviceutil</b>	Esta clase almacena toda la información con relación a las tareas y subtareas que estas contengan.	Modelo
<b>impactolocalserviceutil</b>	Esta clase contiene la información asociada al impacto de cada proyecto.	Modelo

Tabla 2.15 Descripción de clases de diseño.

### 2.4.3 Arquitectura de la información

**Definición de la audiencia:** el portal constará de una audiencia conformada por 3 grupos, los usuarios, el administrador y el avalador.

**Clasificación de la audiencia:**

Usuarios: este grupo determina la mayoría de audiencia interna a la que va encaminado el portal, estando conformado por todos los individuos que interactúen con el portal.

Administrador: este grupo constituirá una minoría, se definen como los instrumentadores del manejo del portal. Esta audiencia está compuesta por usuarios que el directivo de Ciencia y Técnica del Ministerio de Educación Superior defina.

Avalador: este grupo está constituido por aquellas personas definidas por el directivo de la Dirección de Ciencia y Técnica del MES, personal calificado para realizar la tarea de avalar un proyecto o no.

**Necesidades de la audiencia:**

Usuarios: el portal constituirá un medio para obtener información detallada y actualizada concerniente al desarrollo de los proyectos de investigación que el MES maneje, así como noticias relacionadas con los proyectos de investigación que se llevan a cabo y también les valdrá como un medio de aprendizaje. Por otro lado, tendrán la oportunidad de agregar sus investigaciones mostrando el estado de los mismos, obteniendo opiniones y aportes sobre el tema mediante foros, blogs o wikis.

Administrador: tendrá los privilegios para eliminar proyectos y brindar o no los permisos a los avaladores.

Avaladores: tendrá la oportunidad de consultar cada uno de los proyectos de investigación, evaluando los mismos a través de parámetros definidos por ellos, donde decidirá si pueden ser avalados o no.

**Mapa de navegación.**

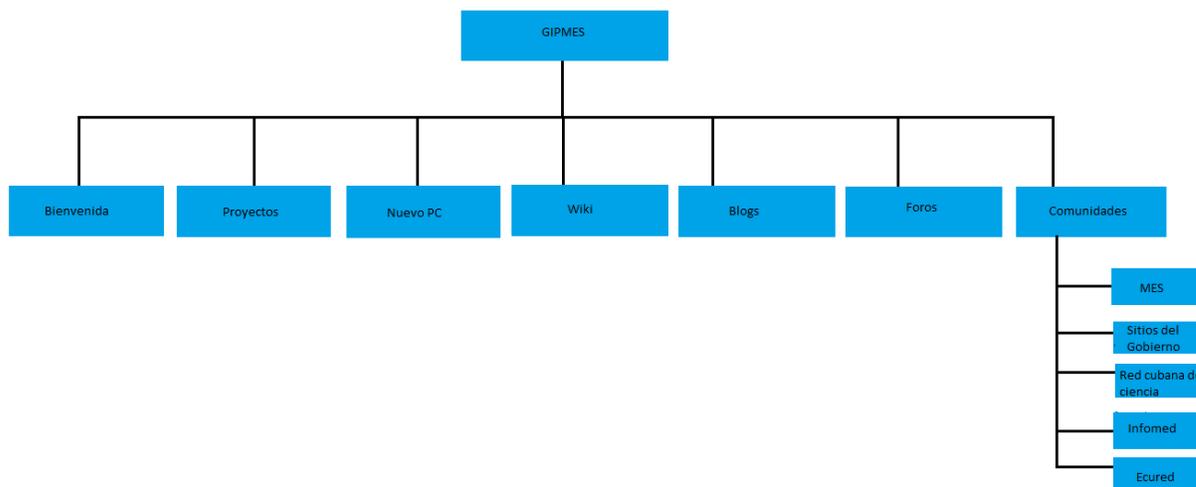


Figura 2.8. Mapa de navegación.

**Estructura del portal.**

El portal para la gestión de la información asociada a la actividad científica del MES está conformado por un conjunto de páginas web, a los cuales tendrán acceso los usuarios de acuerdo a los roles definidos. Cada página del portal estará constituida por una cabecera (*header*), un contenedor de *portlets* y un pie de página (*footer*). La cabecera estará ubicada en la parte superior, conformada del menú de navegación y el logotipo del centro. Contará con un *portlets* para añadir el proyecto y otro para mostrar el proyecto en sí. El pie de página contendrá acceso a otros sitios de interés.

Interfaces del portal.

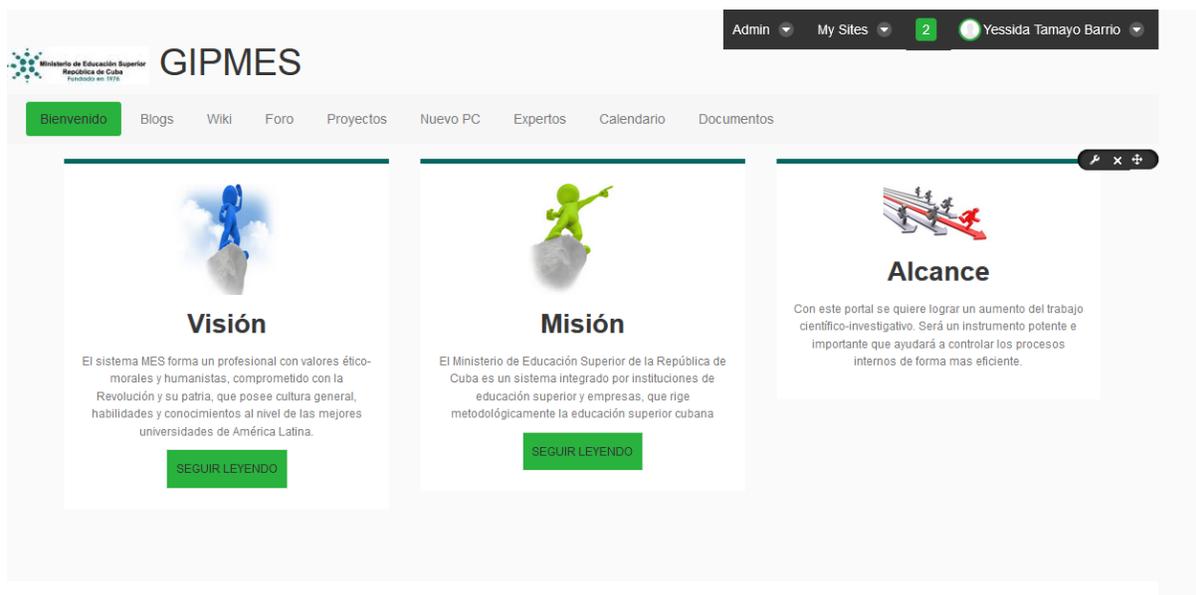


Figura 2.9 Interfaz: bienvenida.

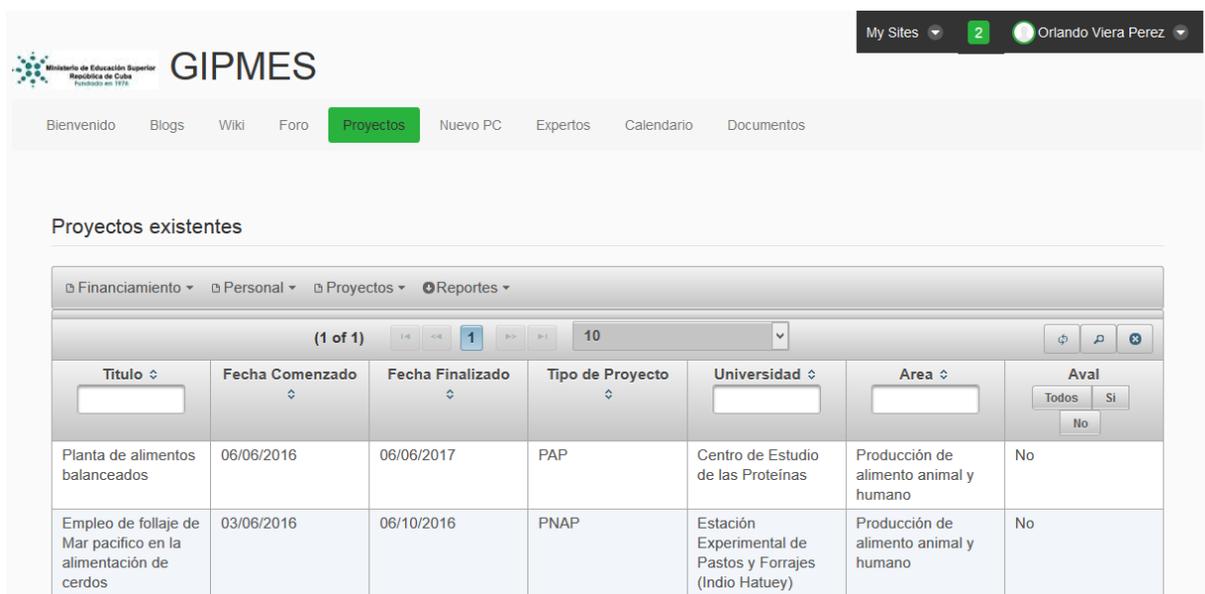


Figura 2.10 Interfaz: listar proyectos.

Ministerio de Educación Superior  
República de Cuba  
Fundado en 1976

**GIPMES**

Admin My Sites 2 Yessida Tamayo Barrio

Bienvenido Blogs Wiki Foro Proyectos Nuevo PC Expertos Calendario Documentos

Blogs  
Add Blog Entry Permissions

Keywords Search

RSS Subscribe

## Crean nuevo proyecto para niños especiales

6/3/16 2:19 PM

Edit Permissions Move to the Recycle Bin

La Universidad de La habana (UH) esta desarrollando un proyecto para niños especiales con el siguiente propósito:

**PRACTICAS DEL LENGUAJE:**

- Plantear situaciones de dialogo en la que todos puedan interactuar verbal y gestualmente, escuchar y ser escuchados.
- Crear espacios para propiciar que emerjan diversas estrategias comunicativas (gestual, oral, corporal).
- Propiciar y estimular el ejercicio de la escucha.
- Propiciar propuestas que involucren a la familia y comunidad (recetas, cuentos, obras de títeres, campañas de prevención)
- Propiciar el aprendizaje desde la escucha y la propia experimentación.

Tag Cloud  
alimento (1) ingeniería informática (1)

Recent Bloggers

**Liliana Rodríguez Martínez**  
Posts: 1  
Stars: 3  
Date: 6/3/16

**Lidia María María Paneque Domínguez**  
Posts: 1  
Stars: 4  
Date: 6/3/16

**Raul Pérez Guevara**  
Posts: 1  
Stars: 4  
Date: 6/2/16

Figura 2.11 Interfaz: blogs.

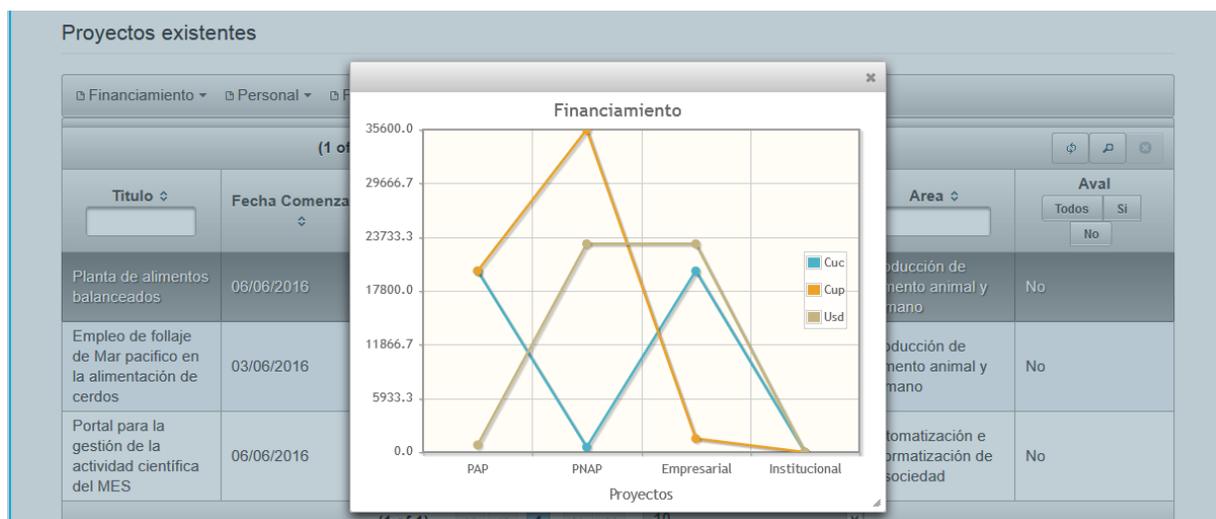


Figura 2.12 Interfaz: gráfica de financiamiento.

Figura 2.13 Interfaz: calendario.

Thread	Flag	Started By	Posts	Views	Last Post	Actions
Crean nuevo proyecto para la gestión de la actividad científica del MES		Laura Suarez Diaz	2	4	Date: 6/6/16 4:00 PM By: Orlando Viera Perez	Actions
quisieran que pusieran en la wiki información acerca del proyecto CA		Ever Leandro Santiesteban Jimenez	1	1	Date: 6/3/16 2:43 PM By: Ever Leandro Santiesteban Jimenez	Actions
Abdominales Inferiores	Waiting for an Answer	Laura Suarez Diaz	4	7	Date: 6/3/16 1:40 AM By: Yaima Martinez Perez	Actions
Sitio de reservacion para comedores de la UCI		Liliana Rodriguez Martinez	3	5	Date: 6/3/16 1:17 AM By: Lidia Maria Maria Paneque	Actions

Figura 2.14 Interfaz: foros.

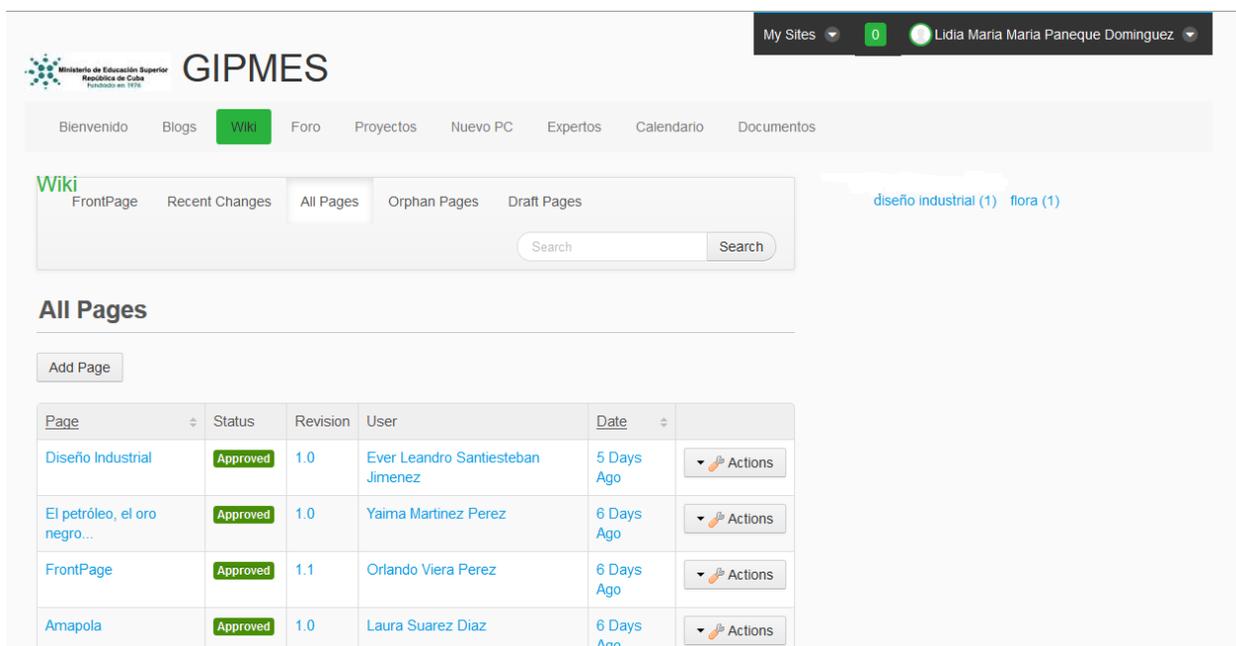


Figura 2.15 Interfaz: wikis.

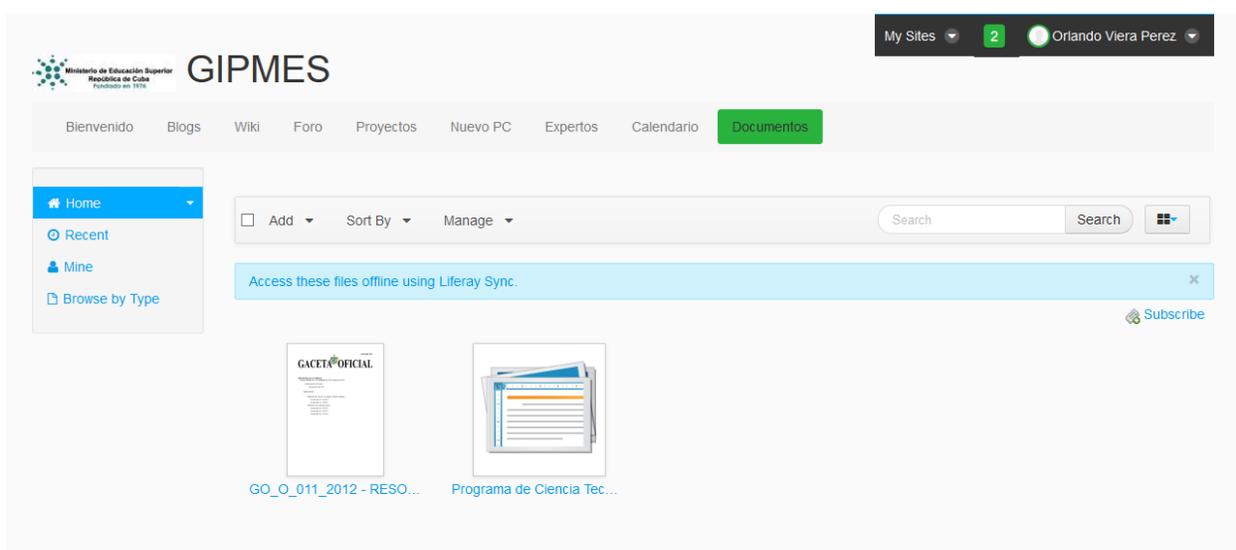


Figura 2.16 Interfaz: publicación y descarga de documentos rectores.

NOMBRE	ENTIDAD	CAT CIENTÍFICA	AÑOS EXPERIENCIA	CORREO	TELEFONO
Alejandro	UCI	Especialista	10	Agomez@correo.uci.cu	78372980
Raúl	Ministerio de Edicación Superior	Máster	6	rpguevara@correo.uci.cu	78373049
Total :2					

Figura 2.17 Interfaz: expertos.

Figura 2.18 Interfaz: crear nuevo proyecto.

### Conclusiones del capítulo.

- Para el correcto cumplimiento de los requisitos definidos, se establecieron parámetros fundamentales para la correcta implementación del portal como fueron la arquitectura, así como los patrones de diseño que facilitan el trabajo.
- Se generaron todos los artefactos de la metodología AUP verificándose la factibilidad de su uso para dar solución a la problemática planteada.

- Se realizó una arquitectura de la información con el fin de conocer todos los usuarios que interactuaran con el sistema, así como sus permisos y necesidades y un mapa de navegación para facilitar la identificación de los principales conceptos incluyendo la relación entre ellos.
- Con el fin de evitar la duplicidad de los datos y garantizar la integridad referencial se diseñó el diagrama entidad relación.
- Se mostraron algunas de las diferentes interfaces que presentará el sistema.

## CAPÍTULO 3 Implementación y pruebas

En el presente capítulo se describe el proceso de implementación de la solución propuesta. Se plasman los casos de pruebas realizados a las historias de usuario correspondientes demostrando de esta forma el cumplimiento de los requerimientos definidos. Por otra parte, se definen las pautas para un correcto despliegue en el Ministerio de Educación Superior.

### 3.1 Estándar de código

Un estándar de código en programación se realiza con el fin de que cualquier persona implicada con el mismo pueda comprender el código y poder trabajar en él.

**Variables:** para estos identificadores haremos uso de la variante lowerCamelCase. Empiezan con minúsculas y si estos identificadores están compuestos por varias palabras las siguientes comenzarán con mayúscula.

**Métodos:**

Los nombres de métodos deben iniciar con un verbo.

Ejemplo:

```
public void addImpacto(...) {...}
```

Los métodos para obtener campos privados en las clases tienen el prefijo "get".

Ejemplo:

```
public List<Proyecto> getFiltered() {...}
```

Los modificadores de campos privados en las clases tienen el prefijo "set".

Ejemplo:

```
public void setProyectos(...) {...}
```

Los obtenedores con el resultado de booleano tienen como nombre un verbo.

Ejemplo:

```
public Boolean getAdministrador(...) {...}
```

**Constantes:**

Las constantes o campos finales son escritos todos con letras minúsculas.

Ejemplo:

```
proyectos= ProyectoLocalServiceUtil.allProyectos();
```

**Clases:**

Para estos identificadores se hace uso de la variante UpperCamelCase. Todas las palabras que componen a dichos identificadores empezarán con mayúscula.

Ejemplo:

```
public class PortletJSF implements Serializable{...}
```

**Comentarios de las clases:**

Al inicio de cada clase se debe describir con un comentario de bloque el propósito de la clase e instrucciones de uso.

Ejemplo:

```
/**
```

```

*
* El método permite obtener un permiso dado el identificador del rol y el
* identificador del portlet.jsf.
*
* @parámetro long idrol
* el primer valor
* @parámetro PortletJSF
* el segundo valor
* @retorna Permiso
*
*/

```

### 3.2 Despliegue del Liferay

Liferay funciona en todos los sistemas operativos, bases de datos, servidores *Lightweight Directory Access Protocol* (LDAP), servidores de aplicaciones. Minimiza la necesidad de inversión en nuevas infraestructuras y apuesta por un conjunto de estándares abiertos. Para su despliegue correctamente se manejan un conjunto de conceptos que el equipo de trabajo luego de su estudio ha puesto en práctica.

**Balanceo de carga:** en la actualidad, debido a la gran demanda de servicios de internet y a la transferencia de información de todo tipo, es incuestionable que los sistemas informáticos deben funcionar de forma ininterrumpida y sin errores en todo momento. El balanceo de carga se basa en un grupo de servidores los cuales cooperan entre ellos para proveer un servicio, incluso al momento de la falla de un componente. Cuando un nodo pierde conectividad, los otros retoman su rol, esto involucra apropiarse de la ip y de los servicios que proveía el nodo que perdió conectividad.

**Nginx:** Nginx es un servidor *web/proxy* inverso ligero y robusto con una infinidad de configuraciones adaptables a nuestras necesidades perteneciente a la familia de *open source*. Es recomendado su uso en sitios donde debe soportar un número elevado de usuarios. Como un *proxy* inverso, puede ser utilizado como un punto único, controlando el acceso de varios servidores de servicios o aplicaciones adicionales, tales como el balanceo de carga. Mediante este se pueden controlar dos categorías de problema: problemas de recursos dentro de la propia Nginx, y también el desarrollo de problemas en otras partes de su infraestructura web (NGINX, 2016).

**Sistema de Autenticación Central CAS:** con el uso de *Central Authentication System* (Sistema de Autenticación Central) se persigue lograr a través de una cuenta única que el usuario acceda desde cualquier lugar, a los servicios informáticos que brinde el sistema y a su vez a los diferentes servicios que ofrecen otros sistemas. Para esto el sistema CAS posee una base de datos donde se almacena la información de identidad de los usuarios (UNICON, 2016).

**Integración CAS + Liferay:** *Single Sign-On (SSO)* es un mecanismo por el cual permite a un usuario autenticarse en diferentes aplicaciones con una sola instancia de identificación. CAS es una de las herramientas SSO, en el sistema se integró el portal del Liferay con el CAS.

**SSO:** inicio de sesión único es un proceso de autenticación de sesión/usuario que permite a un usuario introducir un nombre y una contraseña para poder acceder a múltiples aplicaciones. Estas credenciales de autorización se almacenan en un servidor dedicado a la política SSO, que pasa a lo largo de la credencial de autenticación específica que ha almacenado para cada usuario de manera individual.

Esto permite que el usuario se pueda autenticar en todas las aplicaciones para las cuales se les ha dado derecho y elimina otras indicaciones cuando el usuario cambia de aplicaciones mientras se mantiene en la misma sesión. SSO es útil para documentar el registro y seguimiento de las cuentas de usuario, lo que mejora en gran medida la seguridad de la organización (TECHTARGET, 2016).

**Integración LDAP + Liferay:** el *Lightweight Directory Access Protocol (LDAP)* es un protocolo de servicio de directorio que se ejecuta en una capa por encima de la pila TCP/IP. Proporciona un mecanismo que se utiliza para conectarse, buscar y modificar los directorios de internet. El servicio de directorio LDAP se basa en un modelo cliente-servidor.

Los servicios de directorio juegan un papel importante en el desarrollo de aplicaciones de intranet e internet al permitir el intercambio de información acerca de los usuarios, sistemas, redes, servicios y aplicaciones en toda la red. A modo de ejemplo, los servicios de directorio pueden proporcionar cualquier conjunto organizado de registros, a menudo con una estructura jerárquica, como un directorio de correo electrónico corporativo. En el caso del sistema que se ha desarrollado se hará la integración con el Liferay para poder aprovechar los beneficios que esto aporta.

Cuando configuramos un servidor LDAP con Liferay, podemos importar los diferentes usuarios que se encuentren en el LDAP al servidor Liferay. Esto es muy útil para que cuando los usuarios se identifiquen, el portal automáticamente obtenga los permisos otorgados en el portal, los cuales pueden ser diferente a los que tiene en el LDAP.

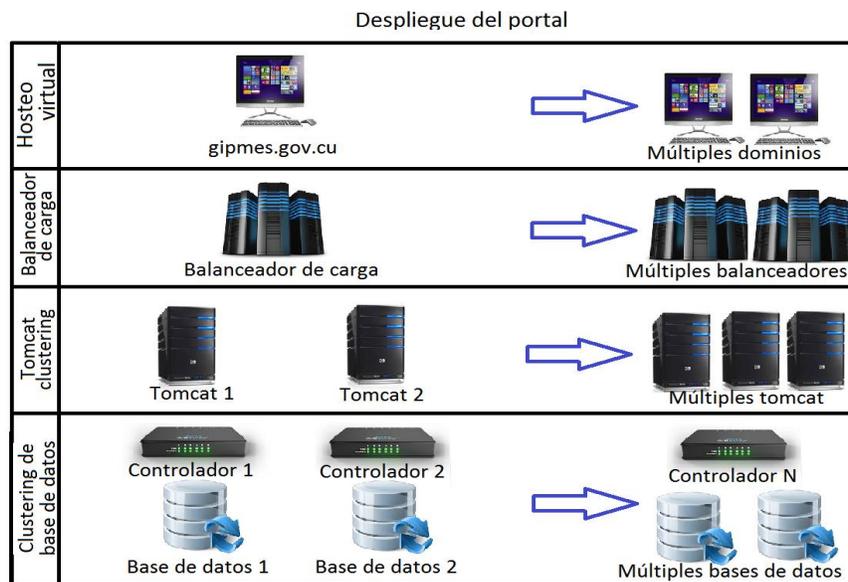


Figura 3.1. Balanceo de carga (LIFERAYS, 2016).

### 3.3 Pruebas

Las pruebas de software, son los procesos que permiten comprobar la calidad de un sistema, identificando posibles fallos de implementación, calidad o usabilidad, son pruebas funcionales dedicadas a examinar en el exterior de lo que se prueba y pruebas de sistemas dedicadas a comprobar los requerimientos no funcionales, donde se evalúa el rendimiento, volumen, carga, seguridad, tensión entre otros aspectos. Se puede definir como una fase dentro del desarrollo de software, donde a través de pruebas se verifican las aplicaciones. Estas están integradas dentro de la ingeniería de software, para determinar el nivel de factibilidad y calidad las aplicaciones son probadas mediante técnicas experimentales usando pruebas de caja negra.

### 3.4 Estrategia de pruebas

Nombre de la sección	Escenarios de la sección	Descripción de la funcionalidad	Flujo Central
SC 1: acceder al sistema.	EC 1.1: se accede correctamente al sistema .	El sistema debe permitir que todo <b>profesor</b> <b>investigador</b> director de proyecto de las universidades y ECTI adscriptas al MES puedan acceder al sistema	1- En el portal Liferay, accionar el botón acceder. 2-Se mostrará una ventana para introducir los datos del usuario. 3-Se accederá a la ventana de bienvenida.

		para la introducción de datos.	
SC 2: crear cuenta.	EC 2.1: se registró correctamente el usuario.	El sistema debe dar la posibilidad de que la persona se inscriba de forma automática mediante la introducción de un nombre de usuario y una clave.	<ol style="list-style-type: none"> <li>1-En el portal Liferay accionar el botón acceder.</li> <li>2-Se muestra una ventana con la opción de registrarse.</li> <li>3-Se muestran los campos a llenar.</li> </ol>
SC 3: crear nuevo proyecto.	EC 3.1: acción realizada satisfactoriamente.	El sistema debe tener una opción "crear un nuevo proyecto" donde el dirigente del proyecto introducirá la información sobre el proyecto según formato que se anexa (anexo No 1). La información de esta opción se podrá guardar siempre que estén llenas todas las solicitudes.	<ol style="list-style-type: none"> <li>1- En el portal Liferay accionar en la pestaña agregar proyecto.</li> <li>2- Aparece el <i>portlets</i> Gestionar Proyecto.</li> <li>3- Introduce los valores y se da a la opción añadir.</li> </ol>
SC 4: modificar proyecto.	EC 4.1: acción actualizada satisfactoriamente.	El sistema debe permitir al usuario modificar los campos de un proyecto que el mismo haya creado.	<ol style="list-style-type: none"> <li>1- En el <i>portlets</i> listar proyecto ir al botón detalles.</li> <li>2- Si eres el creador del proyecto de la opción de modificar.</li> </ol>
SC 5: eliminar proyecto.	EC 5.1: acción eliminada satisfactoriamente.	El sistema debe permitir al usuario eliminar su proyecto	<ol style="list-style-type: none"> <li>1- En el <i>portlets</i> listar proyecto, si es</li> </ol>

		de investigación si este lo desea.	administrador aparece la opción de eliminar. 2- Dar a la opción de eliminar proyecto.
SC 6: mostrar monto de funcionamiento y recursos por proyectos.	EC 6.1: acción realizada satisfactoriamente.	El sistema debe mostrar el monto de financiamiento y recursos por proyectos total del MES.	1- En el <i>portlets</i> listar proyecto aparece un menú con diferentes opciones. 2- Escoger mostrar monto de funcionamiento y recursos. 3- Aparece un modal Windows con la información requerida.
SC 7: mostrar cantidad total de profesores y alumnos vinculados al proyecto.	EC 7.1: acción realizada satisfactoriamente.	El sistema debe permitir mostrar cantidad total de profesores y alumnos vinculados a proyectos.	1-En el <i>portlets</i> listar proyecto aparece un menú con diferentes opciones. 2-Escoger mostrar cantidad total de profesores y alumnos vinculados al proyecto. 3- Aparece un modal Windows con la información requerida.
SC 8: mostrar cantidad de proyectos .	EC 8.1: se mostró correctamente la cantidad de proyectos.	El sistema debe mostrar cantidad de proyectos PAP, PNAP, Empresariales e Institucionales.	1- En el <i>portlets</i> listar proyecto aparece un menú con diferentes opciones. 2- Escoger mostrar cantidad de proyectos. 3- Aparece un modal Windows con la información requerida.
SC 9: acceder desde la ventana principal a un documento pdf con la resolución	EC 9.1: acción realizada correctamente.	El sistema debe permitir acceder desde la ventana principal a un documento pdf con	1- En la ventana de Bienvenida aparece el documento pdf.

<p>44/2012 del CITMA.</p>		<p>la resolución 44/2012 del CITMA y que esta pueda ser cambiada si se modifica (Esto podrá solo introducirse desde el MES).</p>	<p>2- Para acceder accionar en dicho documento.</p>
<p>SC 10: acceder a la relación de programas priorizados por el CITMA.</p>	<p>EC 10.1: acción realizada satisfactoriamente.</p>	<p>El sistema debe permitir acceder desde la ventana principal a la relación de programas priorizados por el CITMA y su contenido (documento Word, esto podrá introducirse en la aplicación solo desde el MES).</p>	<p>1- En la ventana de Bienvenida aparece el documento pdf. 2- Para acceder accionar en dicho documento.</p>
<p>SC 11: realizar comentarios.</p>	<p>EC 11.1: comentario guardado correctamente.</p>	<p>El sistema debe tener en la página un espacio para realizar comentarios.</p>	<p>1- En la página de bienvenida aparece la opción de búsqueda de usuario. 2- Se busca al usuario deseado. 3- Se realiza un comentario en su perfil.</p>
<p>SC 12: listar proyectos.</p>	<p>EC 12.1: se listó correctamente los proyectos existentes.</p>	<p>El sistema debe listar proyectos.</p>	<p>1- Accionar la pestaña Proyectos. 2- Se muestra una lista con cada uno de os proyectos.</p>

SC 13: realizar aval científico.	EC 13.1: acción realizada correctamente.	El sistema debe permitir que el personal calificado evalúe el proyecto de investigación y en dependencia de esto lo avale o no.	1- En el <i>portlets</i> listar proyecto ir al botón detalles. 2- Si eres el avalador te muestra la opción de avalar el proyecto.
----------------------------------	------------------------------------------	---------------------------------------------------------------------------------------------------------------------------------	--------------------------------------------------------------------------------------------------------------------------------------

Tabla3.1 Estrategia de Pruebas.

### 3.5 Casos de prueba

Las celdas de la tabla contienen **V**, **I**, o **N/A**. **V** indica válido, **I** indica inválido, y **N/A** que no es necesario proporcionar un valor del dato en este caso, dado que es irrelevante.

Id del escenario	Escenario	Variable 1 Nombre del usuario.	Variable2 Contraseña.	Respuesta del Sistema	Resultado de la Prueba
EC 1	Acceder al Sistema.	V ytamayo	V asd	Se redirecciona para la página de bienvenida correctamente.	Muestra la página de bienvenida.
		I nulo	V ghj	Notificación de error.	
		V ytamayo	I nulo	Notificación de error.	
		I nulo	I nulo	Notificación de error.	

Tabla3.2 Caso de Prueba: Acceder al Sistema.

### 3.6 Lista de chequeos

Característica de soporte por comprobar.	Ponderación de importancia 1(poco), 5 (fundamental)	Nivel de cumplimiento (0-100)	Justificación
¿Se utilizó como servidor de base de datos PostgreSQL 9.2.4 para el desarrollo del sistema?	4	100	Se utilizó por uno de los más usados actualmente en la Universidad de Ciencias Informáticas. También tiene

			<p>sus licencias liberadas por lo que su uso no es problema y su código fuente se encuentra disponible sin costo alguno.</p>
<p>¿El portlet administrativo es adaptable a los navegadores Moxilla Firefox 46.0.1 , Internet Explorer 11, Chrome 30, Safari u Opera 10?</p>	5	100	<p>El sistema se ejecutó en diferentes entornos de navegación: Moxilla Firefox, Internet Explorer, Chrome, Safari y Opera para darles más facilidades al cliente.</p>
<p>¿Para desarrollar la aplicación se utilizó como servidor web el Apache Tomcat 7.0?</p>	5	100	<p>Se utilizó este servidor ya que posee abundante ayuda y documentación.</p> <p>Es uno de los más usados en el centro CDAE.</p> <p>Cuenta con una comunidad de usuarios y es libre.</p>

Tabla 3.3 Lista de chequeos.

### 3.7 Resultados

Para comprobar que el sistema cumplía con los requerimientos del cliente y por tanto funcionaba correctamente se realizaron dos iteraciones de caja negra basándose en los casos de prueba que se habían definido con anterioridad. Luego de realizadas las iteraciones se pudo corroborar el buen funcionamiento del sistema. En la primera iteración surgieron faltas a la hora de las notificaciones de error y la otra no conformidad encontrada fue respecto a las validaciones de la aplicación. Al realizar la segunda iteración no se encontraron más errores por lo que de esta manera se evaluó cada uno de los casos de prueba de forma satisfactoria.

### 3.8 Pruebas de aceptación

Las pruebas de aceptación se hacen con el objetivo de evaluar el grado de calidad del software con relación a todos los aspectos relevantes para que el uso del producto se justifique (PRUEBASDESFTWARE, 2009). Con su empleo se persigue evaluar la disposición del sistema para su despliegue y posterior uso. Entre los datos que componen una prueba de este tipo están:

- **Iteración:** número de la iteración realizada.
- **Nombre del rasgo a probar:** nombre del rasgo al que se le realiza la prueba.
- **No. del rasgo a probar:** número del rasgo al que se le realiza la prueba.
- **Descripción del rasgo a probar:** descripción breve del rasgo que se prueba.
- **Precondiciones:** condiciones necesarias para poder realizar el caso de prueba y obtener los resultados esperados.

- **Pasos:** pasos lógicos a seguir durante el desarrollo de la prueba para la obtención del resultado esperado.
- **Resultados esperados:** descripción breve de los resultados esperados luego de realizar la prueba.
- **Evaluación:** clasificación de la prueba dependiendo de la comparación del resultado obtenido con el resultado esperado.

Para mostrar la estrategia seguida durante las pruebas de aceptación y los resultados alcanzados, se tomaron tres requisitos a modo de ejemplo. A continuación, se describen las pruebas realizadas.

**Diseño de caso de prueba: crear nuevo proyecto.**

<b>Iteración No.: 1</b>	
<b>Nombre del requisito a probar: crear nuevo proyecto.</b>	<b>Número del requisito a probar: RF 3.1</b>
<b>Descripción del requisito a probar:</b> el sistema debe tener una opción “crear un nuevo proyecto” donde el dirigente del proyecto introducirá la información sobre el proyecto según formato que se anexa (anexo No 1). La información de esta opción se podrá guardar siempre que estén llenas todas las solicitudes.	
<b>Precondiciones:</b> el usuario debe estar previamente registrado y autenticado en el sistema.	
<b>Pasos:</b>	
<ol style="list-style-type: none"> <li>1. El usuario acceder al sistema.</li> <li>2. Ir a la pestaña agregar.</li> <li>3. Llenar los campos correspondientes.</li> </ol>	
<b>Resultados esperados:</b> se agrega el proyecto satisfactoriamente a la lista de proyectos.	
<b>Evaluación:</b> prueba no satisfactoria.	
<b>Observaciones:</b> permite que la fecha final del proyecto sea menos que la fecha inicial.	

*Tabla 3.4 Diseño de caso de prueba: crear nuevo proyecto.*

**Diseño de caso de prueba: modificar proyecto.**

<b>Iteración No.: 1</b>	
<b>Nombre del requisito a probar: modificar proyecto.</b>	<b>Número del requisito a probar: RF 3.2</b>
<b>Descripción del requisito a probarse:</b> muestran los detalles del proyecto a modificar.	
<b>Precondiciones:</b> el usuario debe coincidir con el que creo el proyecto.	

<p><b>Pasos:</b></p> <ol style="list-style-type: none"> <li>1. El usuario accede a la pestaña proyecto.</li> <li>2. Se selecciona el proyecto deseado y se va a detalles.</li> <li>3. Modifica los campos deseados.</li> </ol>
<p><b>Resultados esperados:</b> muestra los cambios efectuados satisfactoriamente en los detalles del proyecto.</p>
<p><b>Evaluación:</b> prueba no satisfactoria.</p>
<p><b>Observaciones:</b> permite a usuarios que no son precisamente los que lo han creado modificar elementos.</p>

Tabla 3.5 Diseño de caso de prueba: modificar proyecto.

<b>Iteración No.: 2</b>	
<b>Nombre del requisito a probar: modificar proyecto.</b>	<b>Número del requisito a probar: RF 3.2</b>
<b>Descripción del requisito a probarse:</b> muestran los detalles del proyecto a modificar.	
<b>Precondiciones:</b> el usuario debe coincidir con el que creo el proyecto.	
<p><b>Pasos:</b></p> <ol style="list-style-type: none"> <li>1. El usuario accede a la pestaña proyecto.</li> <li>2. Se selecciona el proyecto deseado y se va a detalles.</li> <li>3. Modifica los campos deseados.</li> </ol>	
<b>Resultados esperados:</b> muestra los cambios efectuados satisfactoriamente en los detalles del proyecto.	
<b>Evaluación:</b> prueba satisfactoria.	
<b>Observaciones:</b> cumple con la precondición establecida para realizar cualquier tipo de modificación en un proyecto dado.	

Tabla 3.6 Diseño de caso de prueba: modificar proyecto, iteración 2.

**Diseño de caso de prueba: eliminar proyecto.**

<b>Iteración No.: 1</b>	
<b>Nombre del requisito a probar: eliminar proyecto.</b>	<b>Número del requisito a probar: RF 3.3</b>
<b>Descripción del requisito a probarse:</b> permite eliminar un proyecto de la base de datos.	
<b>Precondiciones:</b> solo puede eliminar un proyecto el usuario que tenga el rol de administrador.	

<b>Pasos:</b> <ol style="list-style-type: none"> <li>1. El usuario accede a la pestaña proyecto.</li> <li>2. Se selecciona el proyecto a eliminar y se da clic en la opción.</li> </ol>
<b>Resultados esperados:</b> elimina el proyecto de la base de datos.
<b>Evaluación:</b> prueba satisfactoria.
<b>Observaciones:</b> -

Tabla 3.7 Diseño de caso de prueba: eliminar proyecto.

### 3.9 Pruebas de rendimiento

Las pruebas de rendimiento y carga son de gran importancia, una mala configuración del servidor web, o de la red donde correrá el sistema puede influenciar negativamente en el uso del mismo (BANCHOFF, 2008). El equipo de trabajo seleccionó como herramienta a utilizar JMeter para la etapa de testeo de rendimiento del servidor web para el sitio en desarrollo.

Características del ambiente de pruebas.

Hardware	
Tipo de procesador	Core I3 2da generación
Memoria del sistema	4 gb RAM
Tipo de red	LAN
Software	
Tipo de servidor web	Apache Tomcat 7.0
Memoria Máxima	4 gb RAM
Máximo de hilos concurrentes	50/páginas
Plataforma	Windows
Servidor de Base de Datos	Postgres SQL 9.2.4

Tabla 3.8 Características del ambiente de prueba.

El resumen del reporte permite visualizar los resultados del test realizado, en una tabla. Los datos que se presentan son:

- *Label:* etiqueta de la muestra.
- # Muestras: cantidad de hilos utilizados para la URL.
- Media: tiempo promedio en milisegundos para un conjunto de resultados.
- Min: tiempo mínimo que demora un hilo en acceder a una página.
- Máx: tiempo máximo que demora un hilo en acceder a una página.

- Rendimiento: rendimiento medido en los requerimientos por segundo/minuto/hora.
- Kb/sec: rendimiento medido en Kbyte por segundo.
- Media en bytes: tamaño medio de respuesta al servidor (en bytes) (BANCHOFF, 2008).

Label	# Muestras	Media	Mín	Máx	Std. Dev.	% Error	Rendimiento	Kb/sec	Avg. Bytes
Comprobando Bienvenida	50	8889	3509	21679	4913,51	0,00%	2,3/sec	112,09	50648,2
Comprobando Añadir	50	8225	1981	14702	3895,30	0,00%	2,5/sec	70,42	28452,7
Comprobando wiki	50	3966	290	8939	1737,07	0,00%	2,7/sec	75,70	28427,2
Comprobando Proyectos	50	2797	1053	5448	1012,80	0,00%	4,0/sec	112,25	28466,9
Comprobando foros	50	2087	330	3824	1030,59	0,00%	5,3/sec	148,06	28434,3
Comprobando blog	50	1874	281	3782	896,18	0,00%	5,9/sec	164,38	28426,0
TOTAL	300	4640	281	21679	3961,34	0,00%	9,9/sec	311,94	32142,5

Tabla 3.9. Resumen de reporte (CAULES, 2014).

*Agregate Graph*: este componente es similar al anterior, pero permite obtener resultados más precisos. Utiliza más memoria, ya que calcula la mediana y la línea al 90%, la cual requieren que todos los datos estén almacenados. Los datos que se presentan son:

- URL: etiqueta de la muestra
- #Muestras: cantidad de hilos utilizados para la URL.
- Media: tiempo promedio en milisegundos para un conjunto de resultados.
- Mediana: valor en tiempo del percentil 50.
- Línea de 90%: máximo tiempo utilizado por el 90% de la muestra, al resto de la misma le llevó más tiempo.
- Min: tiempo mínimo de la muestra de una determinada URL.
- Max: tiempo máximo de la muestra de una determinada URL.
- %Error: porcentaje de requerimientos con errores.
- Rendimiento: rendimiento medido en los requerimientos por segundo / minuto / hora.
- KB/sec: rendimiento medido en Kbytes por segundo(BANCHOFF, 2008).

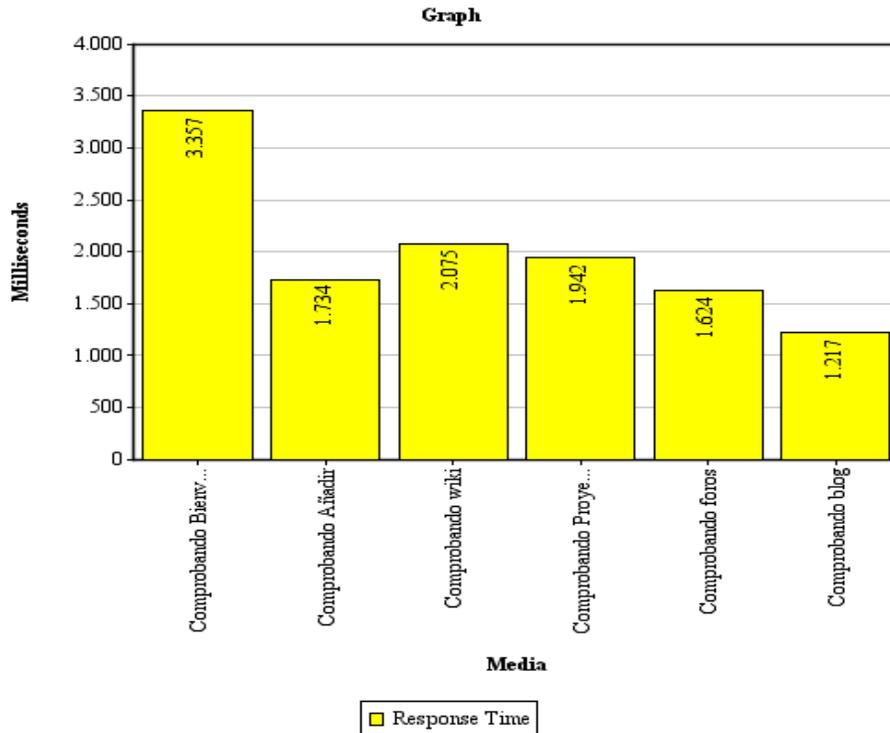


Gráfico3.1. Gráfico agregado.

El gráfico de resultados (siguiente gráfico) por su parte permite visualizar gráficamente los siguientes datos: media, mediana, dispersión y el rendimiento (representando como el número actual de requerimientos/minutos que el servidor maneja) (BANCHOFF, 2008).

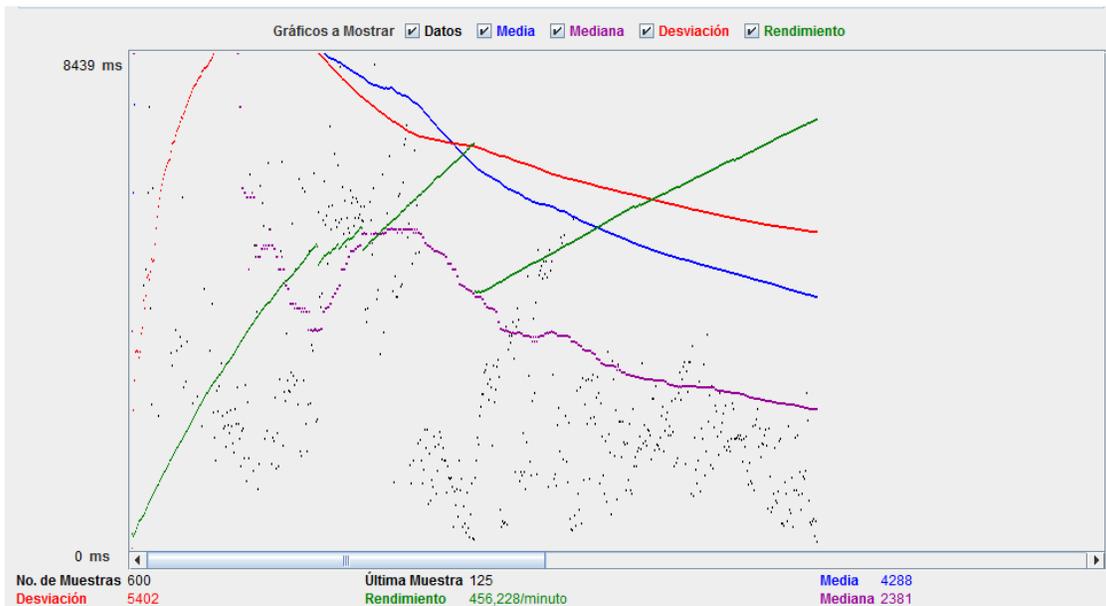


Gráfico3.2 Gráfico de resultados.

### **Resultados obtenidos.**

Para el ambiente de trabajo antes descrito se obtuvieron los siguientes resultados:

La prueba realizada para 50 hilos por página arrojó los siguientes resultados: El tiempo máximo invertido por petición fue de 21s (columna Máx), con un mínimo de tiempo invertido por una petición (columna Min) de 0,281s y una media de tiempo invertido por una petición (columna Media) de 4s. El rendimiento, número de peticiones procesadas en una unidad de tiempo, que puede expresarse segundo, minutos y horas fue de 9,9/sec.

### **Conclusiones parciales**

- Se pudo comprobar a través de los resultados alcanzados en las pruebas ejecutadas que se obtuvo un software con la calidad requerida.
- Se probaron cada uno de los requisitos establecidos por el cliente logrando a través de un conjunto de iteraciones que los mismos funcionaran correctamente.
- Se logró una interfaz intuitiva para el usuario y de fácil manejo.
- Se realizó una estrategia de despliegue con el fin de poner el sistema a la disposición del cliente.

## **CONCLUSIONES**

- Se realizó un estudio de los principales sistemas relacionados con la gestión de información, así como de portales empresariales lo que sirvió de gran ayuda para seleccionar la plataforma empresarial a utilizar para el desarrollo del portal para la gestión de la información asociada a la actividad científica del Ministerio de Educación Superior.
- Mediante la adecuada selección y empleo de métodos, tecnologías y herramientas, se desarrolló para la Dirección de Ciencias y Técnicas del MES, un portal para la gestión de la información asociada a la actividad científica, que mejora el control de la información que es manipulada por el directivo, logrando satisfacer las necesidades diagnosticadas al principio de la investigación.
- Se propuso una arquitectura de despliegue para un escenario de alta disponibilidad.
- A través de técnicas y herramientas asociadas a la ingeniería de software se le aplicaron una serie de pruebas al sistema tanto funcionales como no funcionales, las cuales permitieron ofrecer un producto con la calidad requerida.

## **RECOMENDACIONES**

Para darle continuidad al presente trabajo se recomienda lo siguiente:

- Automatizar el proceso de avalar, donde no solo se dé la oportunidad de ejecutar esta acción, sino que le brinde al mismo un porcentaje de aceptación para decir si se avala o no.
- Se recomienda fortalecer y encaminar los esfuerzos en el campo académico a fin de obtener más y mejores soluciones empresariales acorde con el entorno dinámico altamente competitivo.
- Incorporar una funcionalidad que permita la existencia de artículos que pueda generar un proyecto con sus respectivas evidencias.

## REFERENCIAS BIBLIOGRÁFICAS

- ARNAL, D. M. Conceptos de web 2.0 y biblioteca 2.0: origen, definiciones y retos para las bibliotecas actuales. 2007, nº p. 12.
- BANCHOFF, C. M. T. Usando Jmeter para pruebas de rendimiento. 2008, nº Disponible en: {diaz, cbanchoff, anahi, valeria}@linti.unlp.edu.ar.
- BENITO, I. *Los 5 mejores gestores de contenido (CMS) Open Source en 2015 de 2016*]. Disponible en: <http://algoentremanos.com/los-5-mejores-gestores-de-contenido-cms-open-source-en-2015/>.
- CAPTERRA. *Top Portal Software Products de 2016*]. Disponible en: <http://www.capterra.com/portal-software/>.
- CAULES, C. Á. *Introducción a JMeter y pruebas de carga de 2106*]. Disponible en: <http://www.arquitecturajava.com/introduccion-jmeter-y-pruebas-de-carga/>.
- COMMONS, C. *Mapa visula de la web 2.0. 2007*, Disponible en: <http://www.internality.com/web20/>.
- D, M. S. *Prospectiva de los portales corporativos. 2007, nº*
- FLORES, L. E. *Metodologías ágiles. "Proceso Unificado Agil". de 2016*]. Disponible en: [http://www.academia.edu/7894130/METODOLOGIAS\\_AGILES\\_AUP](http://www.academia.edu/7894130/METODOLOGIAS_AGILES_AUP).
- FOUNDATION, T. A. S. *Apache Tomcat de 2016*]. Disponible en: <http://tomcat.apache.org/>.
- FOUNDATION, T. A. S. *Maven 3.1.0 de 2016*]. Disponible en: <https://maven.apache.org/docs/3.1.0/release-notes.html>.
- FOUNDATION, T. E. *Eclipse de 2016*]. Disponible en: <https://eclipse.org/>.
- GÓMEZ, J. C. G. *Portales de Internet: Concepto, tipología básica y desarrollo. 2001, nº*
- GUILLERMO STORTI, G. R., GABRIEL CAMPODÓNICO. *Bases de Datos, Modelo Entidad Relación. 2007, nº Disponible en:* [http://www.belgrano.esc.edu.ar/matestudio/carpeta\\_de\\_access\\_introduccion.pdf](http://www.belgrano.esc.edu.ar/matestudio/carpeta_de_access_introduccion.pdf).
- HIBERNATE. *Hibernate ORM de 2016*]. Disponible en: <http://hibernate.org/orm/>.
- LIFERAY. *Liferay de 2016*]. Disponible en: <https://www.liferay.com/es/>.

- LIFERAYS. *Liferay load balancer*. 2016, Disponible en: <https://www.google.com/cu/search?q=liferay+load+balancer&hl=es-419&biw=1366&bih=655&site=webhp&source=lnms&tbn=isch&sa=X>.
- MAYOR, A. C. *CMS, LMS y LCMS. Definición y diferencias* de 2016]. Disponible en: <http://www.centrocp.com/cms-lms-y-lcms-definicion-y-diferencias/>.
- MES. *Ministerio de educación superior* de 2015]. Disponible en: [www.mes.gob.cu](http://www.mes.gob.cu).
- NGINX. *Nginx* de 2016]. Disponible en: <http://nginx.org/en/>.
- NIETO, C. G. Á. y ALVARADO, E. F. S. *Ventajas y desventajas de la Tecnología RIA (Rich Internet Applications)*. UNIVERSIDAD DEL AZUAY, 2011.
- ORACLE. *Compositeview* de 2016]. Disponible en: <http://www.oracle.com/technetwork/java/compositeview-137722.html>.
- OSCAR DÍAZ, A. I., MAIDER AZANZA, FELIPE M. VILLORIA. *MODELADO DE LA AGREGACIÓN DE PORTLETS POR MEDIO DE STATECHARTS*. 2006, n°
- PAVÓN, J. *Estructura de las Aplicaciones Orientadas a Objetos. El patrón Modelo-Vista-Controlador (MVC)*. En *Estructura de las Aplicaciones Orientadas a Objetos. El patrón Modelo-Vista-Controlador (MVC)*. 2008.
- PELÁEZ, J. *Arquitectura basada en Capas*. 2009, Disponible en: <http://www.juanpelaez.com/geek-stuff/arquitectura/arquitectura-basada-en-capas/>.
- PRUEBASDESOFTWARE. *Gestión de Calidad y Pruebas de Software* de 2016]. Disponible en: <http://www.pruebasdesoftware.com/pruebadeacceptacion.htm>.
- ROSAS, J. E. S. *Porque usar Liferay para gestionar sitios web* de 2016]. Disponible en: <http://blogs.antartec.com/opensource/2010/10/razones-para-usar-liferay-como-cms/>.
- SCARMACIO, F. *Instalación del Liferay*. 2010, vol. 2016, Disponible en: <http://francescoscamarcio.com/2010/12/13/instalacion-de-liferay/>.
- SOURCE, O. *Creating Service layer in Service Builder in Liferay* de 2016]. Disponible en: <http://www.opensource-techblog.com/2013/03/creating-service-layer-in-service.html>.
- TECHTARGET. *Single sign-on (SSO)* de 2016]. Disponible en: <http://searchsecurity.techtarget.com/definition/single-sign-on>.

TEDESCHI, N. *¿Qué es un Patrón de diseño?* de 2016]. Disponible en:  
<https://msdn.microsoft.com/es-es/library/bb972240.aspx>.

UNICON. *Central Authentication Service (CAS)* de 2016]. Disponible en:  
<https://www.unicon.net/opensource/cas>.

## ANEXOS

## Casos de prueba.

Id del escenario	Escenario	Variable 1 nombre	Variable 2 2donombre	Variable 3 Nombre de usuario	Variable 4 Catchap	Respuesta del Sistema	Resultado de la Prueba
EC 2	Crear cuenta.	V Yessida	N/A nulo	V ytamayo	V 4567	Muestra mensaje de notificación y da una contraseña. Al introducirla te redirecciona para que la cambies, al hacerlo te redirecciona nuevamente para proteger la cuenta.	Te muestra la página de bienvenida.
		I nulo	N/A María	V lpdominguez	V 8769	Muestra un mensaje de notificación de error.	Mantendrá al usuario en la misma página hasta que el mismo llene los campos correctamente.
		V Raúl	N/A nulo	I nulo	V 7214		
		V Brian	N/A nulo	V bmartinez	I 7509		

Tabla 3.10 Caso de prueba: crear cuenta.

Id del escenario	Escenario	Variable 1 Título	Variable2 Línea	Variable3 Objetivos generales	Variable 4 Tareas	Respuesta del Sistema	Resultado de la Prueba
EC 3	Crear proyecto.	V Tablero de Control para la Dirección de Posgrado de la Universidad de las Ciencias Informáticas	V Investigación de postgrado.	V Desarrollar componentes desplegables en la web que faciliten el diseño y creación de un Tablero de Control el cual se contribuya a una efectiva toma de decisiones en la dirección del proceso de formación posgraduada en la UCI.	V Diseño de la arquitectura del sistema.	Guarda correctamente los datos.	Aparecerá un nuevo proyecto en la Base de datos, mostrándose en una lista.
		I nulo	V Producción alimenticia.	V Desarrollar una eficiente planificación en la distribución de alimentos.	V Planificación de distribución de alimentos.	Muestra un mensaje de notificación de error.	
		V Módulo para la automatización de artefactos en la	I nulo	I nulo	V Realizar un estudio del arte.		

		disciplina de Pruebas				
		V Portal para la gestión de proyectos de investigación.	I nulo	V Desarrollar un sistema que permita gestionar los proyectos de investigación científica asociados al MES.	I nulo	

Tabla 3.11 Caso de prueba: crear proyecto.

<b>Id del escenario</b>	<b>Escenario</b>	<b>Variable 1 Título</b>	<b>Variable2 Línea</b>	<b>Variable3 Objetivos generales</b>	<b>Variable4 Tareas</b>	<b>Respuesta del Sistema</b>	<b>Resultado de la Prueba</b>
EC 3	Modificar proyecto.	V Tablero de Control para la Dirección de Posgrado de la Universidad de las Ciencias Informáticas	V Investigación de postgrado.	V Desarrollar componentes desplegables en la web que faciliten el diseño y creación de un Tablero de Control el cual se contribuya a una efectiva toma de decisiones en la dirección del	V Diseño de la arquitectura del sistema.	Guarda correctamente los cambios.	En el proyecto modificado aparecen los nuevos cambios cuando consultamos la lista de proyectos.

				proceso de formación			
				posgraduada en la UCI.			
		I nulo	V Producción alimenticia humana y animal.	V Desarrollar una eficiente planificación en la distribución de alimentos.	V Planificación de distribución de alimentos.	Muestra mensaje notificación error.	un de de

Tabla 3.12 Caso de prueba: modificar proyecto.

Id del escenario	Escenario	Variable 1 Comentario	Respuesta del Sistema	Resultado de la Prueba
EC 11.	Realizar comentarios.	V Muy buena investigación.	Guarda el comentario.	Muestra el comentario en la página.
		I nulo	Muestra una notificación de error.	

Tabla 3.13 Caso de prueba: realizar comentario.