

Universidad de las Ciencias Informáticas

Facultad 6



**Sistema de gestión de información del Centro de Innovación
y Calidad de la Educación de la Universidad de las Ciencias
Informáticas**

Autor:

Ilianet González León

Tutora:

Ing. Yordany Piñeiro Gómez

La Habana, julio de 2016

“Año 58 de la Revolución”

Declaración de autoría

Por este medio declaro que yo, Ilianet González León, con carné de identidad 92062033954, soy la autora principal del trabajo titulado “Sistema de gestión de información del Centro de Innovación y Calidad de la Educación de la Universidad de las Ciencias Informáticas”, y autorizo a la Universidad de las Ciencias Informáticas (UCI) y al Centro de Innovación y Calidad de la Educación (CICE), a hacer uso de la misma en su beneficio, así como los derechos patrimoniales con carácter exclusivo.

Para que así conste firmo la presente declaración jurada de autoría en La Habana a los ____ días del mes de _____ del año 2016.

Ilianet González León

Firma del Autor

Ing. Yordanys Piñeiro Gómez

Firma del Tutor

Tutor: Ing. Yordanys Piñeiro Gómez

Graduado de Ingeniería en Ciencias Informáticas en la Universidad de las Ciencias Informáticas en Julio de 2007.

Categoría docente: Asistente

Profesor de las asignaturas Ingeniería de Software I y II, Gestión de Software e Introducción a las Ciencias Informáticas.

Centro de Trabajo: Universidad de las Ciencias Informáticas (UCI). Centro de Innovación y Calidad de la Educación (CICE).

Cargo: Profesora. Vicerrectoría de Formación.

Dirección: Carretera a San Antonio, Km 2 ½, Reparto Torrens, La Lisa, La Habana. Universidad de las Ciencias Informáticas (UCI), Edificio: 27, Apto: 201.

Teléfono Oficina: +53 – 7 – 837 – 2522.

Correo electrónico: ypineirog@uci.cu

Resumen

Los centros productivos y de investigación que forman parte de la Universidad de las Ciencias Informáticas generan grandes cúmulos de información. Al mismo tiempo, los profesores, estudiantes y trabajadores demandan fuentes informativas y de expresión que les permitan mantenerse actualizados de temáticas que les sean de interés para su desarrollo profesional. El Centro de Innovación y Calidad de la Educación, tiene como misión contribuir a elevar la calidad de los procesos principales de la institución y el desarrollo sistemático de investigaciones en las Ciencias Pedagógicas y de la Educación. Este centro cuenta con un portal web que se ha mantenido inactivo por más de tres años, desarrollado con tecnologías obsoletas y no responde a la dirección estratégica definida en la actualidad. No se dispone de un espacio propio por el cual se divulguen los servicios que se ofrecen y el procesamiento de la información asociada a ciencia, técnica e innovación se realiza de forma manual, lo cual hace que el proceso se torne lento, engorroso y no se disponga en todo momento de información actualizada y fiable. La propuesta de solución consiste en un sistema de gestión de información que promueva los servicios que se brindan en el Centro de Innovación y Calidad de la Educación para incrementar la centralidad, seguridad, disponibilidad y oportunidad de la información que allí se maneja. Para desarrollar este sistema se eligió el Proceso Unificado Abierto como metodología de desarrollo de software, Symfony2 como marco de trabajo y PostgreSQL como gestor de base de datos.

Palabras claves: información, gestión de la información, sistema de gestión de la información.

Abstract

Production and research centers that are part of the University of Computer Science generate large clusters of information. At the same time, teachers, students and workers demand information and expression that enable them to keep abreast of topics of interest to them for their professional development sources. The Center for Innovation and Quality of Education, has the mission of contribute to raising the quality of the main processes of the institution and the systematic development of research in Pedagogical and Educational Sciences. This center has a web site that has been inactive for more than three years, developed with obsolete technologies and does not respond to the strategic direction defined today. There is no own space by which the services offered and information processing associated with science, technology and innovation is done manually, which makes the process becomes slow, cumbersome and not available at all times to date and reliable information. The proposed solution consists of an information management system that promotes the services provided at the Center for Innovation and Quality of Education to increase the centrality, security, availability and timeliness of information handled there. To develop this system Open Unified Process was chosen as software development methodology, Symfony2 as a framework and PostgreSQL as database manager.

Keywords: *information, information management, information management system.*

Introducción	1
Capítulo 1. Fundamentación Teórica	7
1.1 El proceso de gestión de la información. Integridad y disponibilidad de la información	7
1.2 Caracterización del proceso de gestión de la información en el Centro de Innovación y Calidad de la Educación.....	9
1.3 Análisis de las soluciones existentes	11
1.4 Tendencias y tecnologías actuales	15
1.4.1 Metodología de desarrollo de software Proceso Unificado Abierto (OpenUP)	16
1.4.2 Lenguaje unificado de modelado (UML 2.0)	17
1.4.3 Herramienta Case Visual Paradigm for UML 8.0	17
1.4.4 Lenguajes empleados en la programación	18
1.4.4.1 PHP 5.6.....	18
1.4.4.2 HTML 5	18
1.4.4.3 CCS 3	19
1.4.4.4 JavaScript v1.2.....	19
1.4.5 Marco de trabajo	20
1.4.5.1 Symfony v2.8	20
1.4.5.2 Bootstrap v3.1	20
1.4.5.3 JQuery v1.10.2	21
1.4.6 Entorno de Desarrollo Integrado (IDE) PHP Storm 7.4.....	21
1.4.7 Servidor Web Apache 2.4.4.....	23
1.4.8 Sistema gestor de base de datos PostgreSQL v9.4.5	24
Conclusiones parciales.....	25
Capítulo 2. Análisis del sistema	26
2.1 Requisitos de Software	26
2.1.1 Especificación de Requisitos Funcionales.....	26
2.1.2 Especificación de Requisitos no Funcionales	33
2.2 Modelación del Sistema.....	35
2.2.1 Descripción de los actores que interactúan con el sistema.....	35
2.2.2 Estructuración del sistema	36
2.2.3 Descripción textual de los Casos de Uso del Sistema	38
Conclusiones parciales.....	47

Capítulo 3. Descripción de la solución propuesta.....	48
3.1 Arquitectura de Software	48
3.1.1 <i>Estilo de Arquitectura</i>	49
3.1.2 <i>Patrón arquitectónico Modelo-Vista-Controlador</i>	49
3.1.3 <i>Vista lógica</i>	51
3.1.4 <i>Diagrama de clases del diseño</i>	52
3.1.5 <i>Patrones del Diseño</i>	53
3.2 Diseño de la Base de Datos.....	55
3.3 Modelo de despliegue.....	57
Conclusiones parciales.....	58
Capítulo 4. Implementación y Pruebas.....	59
4.1 Diagrama de componentes.....	59
4.2 Estándares de codificación	60
4.3 Tratamiento de errores	61
4.3.1 <i>Tratamiento de errores del lado del cliente</i>	61
4.3.2 <i>Tratamiento de errores del lado del servidor</i>	62
4.4 Pruebas	62
4.4.1 <i>Pruebas de caja negra</i>	63
4.4.2 <i>Pruebas de rendimiento</i>	64
4.4.3 <i>Valoración de la contribución lograda</i>	65
Conclusiones parciales.....	67
Conclusiones	68
Recomendaciones	69
Bibliografía.....	70

Índice de tablas y figuras

Índice de Tablas

Tabla 1 Lista de IDE más mencionados. [tomado de (Okeke Oluchukwu, 2015)]	21
Tabla 2 Descripción de los actores del sistema	35
Tabla 3 Descripción textual del Caso de Uso Gestionar Eventos	38
Tabla 4 Prueba de rendimiento al SGICICE	65

Índice de Figuras

Figura 1 Uso Personal de IDE y editores de PHP. [tomado de (Okeke Oluchukwu, 2015)]	22
Figura 2 Uso en Negocios de IDE y editores de PHP. [tomado de (Okeke Oluchukwu, 2015)]	23
Figura 3 Estructura del sistema a través de paquetes	36
Figura 4 Diagrama de Casos de Uso del Sistema	37
Figura 5 Funcionamiento interno de Symfony con el patrón MVC	50
Figura 6 Vista lógica extendida	51
Figura 7 Vista Lógica de la aplicación	52
Figura 8 Diagrama de Clases del Diseño del Caso de Uso Gestionar Evento	53
Figura 9 Diagrama entidad-relación	56
Figura 10 Diagrama de Despliegue	57
Figura 11 Diagrama de Componentes del Caso de Uso Gestionar Evento	59
Figura 12 Utilización de los estándares de codificación	61
Figura 13 Campos en blanco	61
Figura 14 Autocompletamiento	62
Figura 15 Gráfica de No Conformidades	64

Introducción

En el contexto mundial globalizado, altamente competitivo y con transformaciones tecnológicas de amplio impacto, en el cual las necesidades, demandas y aspiraciones sociales imperan cada día, el interés por la calidad de los servicios educativos se ha incrementado de manera considerable. Se requiere que las instituciones escolares sean más efectivas en el cumplimiento de la misión educativa para las que han sido creadas, lo que ha promovido el desarrollo y consolidación de diversas estrategias de mejora, una de ellas la innovación.

Según (Cáceres, y otros, 2010) la Educación Superior se enfrenta a una serie de desafíos en un mundo que se transforma, esto conlleva a redefinir muchas de sus tareas, en especial aquellas que se relacionen con las necesidades de la sociedad en materia de aprendizaje y superación continua. Las universidades destacan en el empeño por garantizar la educación continua y el posgrado, desarrollando habilidades que sirvan como base para la creación científica. Por esta razón, las actividades de capacitación, grado y posgrado, se convierten en eslabones fundamentales del proceso de formación integral de estudiantes y profesores.

Cuba, no está exenta de los retos trazados, tal es el caso de la Universidad de las Ciencias Informáticas (UCI), creada en el 2002, por idea del Comandante Fidel Castro Ruz, quien afirmó que debía ser una universidad innovadora, de excelencia científica, académica y productiva, que formara de manera continua profesionales integrales comprometidos con la patria, soporte de la informatización del país y la competitividad internacional de la industria cubana del software. La UCI cuenta con el Centro de Innovación y Calidad de la Educación (CICE), el cual tiene como misión: contribuir a elevar la calidad de los procesos principales de la institución, mediante la formación permanente de la comunidad universitaria en el pre y el posgrado y el desarrollo sistemático de investigaciones en las Ciencias Pedagógicas y de la Educación. Desde el propio surgimiento del centro se desarrolla y fomenta la innovación y la investigación, para apoyar la superación permanente y eficaz de los recursos humanos.

En la actualidad el referido centro estructura su trabajo en tres grupos (Formación Pedagógica, Innovación Pedagógica e Investigación, Calidad y Acreditación). El primero de ellos se encarga de orientar metodológicamente y controlar el cumplimiento, en lo referente a la actividad de pregrado de la disciplina Formación Pedagógica del Plan de estudio del Ingeniero en Ciencias Informáticas. De igual forma es el

Introducción

responsable de orientar metodológicamente y controlar el cumplimiento de la actividad de posgrado del centro, además de organizar, registrar, custodiar y mantener actualizada todas las informaciones relativas a la formación postgraduada y la gestión de la calidad educativa que ejecuta el centro de estudios. El segundo debe orientar metodológicamente y controlar el cumplimiento de las actividades de innovación, coordinar los equipos de trabajo para estudiar, organizar, orientar metodológicamente y controlar, los indicadores y formas de gestión de la innovación y el desarrollo científico en el centro. El tercero se dedica a realizar estudios de calidad y desarrollar las estrategias de acreditación para la carrera.

(Lebrún 2005) afirma que, dado que las empresas necesitan mostrar sus productos o servicios a todos sus clientes, se crean los portales web, que su función principal es: servir de puerta de entrada única para ofrecer al usuario, de forma fácil e integrada, el acceso a los recursos y servicios que brinda, incluyendo buscadores, enlaces, foros. De esta manera, queda publicada, la imagen de la empresa, lo cual es aplicable a todos los contextos (instituciones escolares, centros de investigación y de producción). Teniendo en cuenta esta premisa en el año 2005 se creó un portal web para el CICE que respondía a la estructura, misión, objetivos y dirección estratégica característica del centro alusivo. En el año 2013 se realizaron un conjunto de transformaciones que hacían que el portal web que se utilizaba no fuera expresión del trabajo e intereses que se perseguían, desarrollado con tecnologías que ya habían quedado obsoletas y no tenía soporte, por lo que se decidió cerrarlo a la visión pública.

En el centro se organizan, coordinan y ejecutan cursos de posgrado, entrenamientos, diplomados, maestrías y doctorados. En la actualidad no se dispone de un espacio propio y centralizado a través del cual se puedan divulgar todos estos servicios; para lo cual se deben auxiliar de diversos medios como: la intranet, los portales web de las facultades, plegables de escuela de invierno y verano. Unido a lo anterior, los aspectos que identifican el trabajo del CICE (misión, visión, dirección estratégica, estructura, documentos normativos, resultados fundamentales) no son conocidos por la poca visibilidad que existe a nivel institucional, nacional e internacional como consecuencia de lo expresado con anterioridad.

De igual forma, el procesamiento de toda la información de ciencia, técnica e innovación se torna complejo y lento, pues se realiza de forma manual a través de ficheros Excel que no se encuentran automatizados, lo cual atenta contra la integridad de los datos, la constante actualización de la información y el aumento significativo del tiempo dedicado a este fin. También se lleva un control de las revistas asociadas a las Ciencias Pedagógicas y de la Educación, así como de otras temáticas generales orientadas a diferentes

Introducción

perfiles de formación y de los eventos a desarrollar en el año en curso. Lo anterior se realiza a través de ficheros Excel, que son circulados solo a los trabajadores del centro. Dada la cantidad de datos que se almacenan no se pueden generar de forma rápida reportes que se necesiten en un momento determinado, además de que ocurre una mayor probabilidad de errores en los informes finales, ya que depende de la habilidad que tenga una persona para realizar el procesamiento que se necesita para obtener los resultados finales.

Actualmente, el proceso de gestión de la información asociada a los servicios que se brindan en el Centro de Innovación y Calidad de la Educación, presenta las siguientes deficiencias:

- La integridad de la información está comprometida por posibles errores humanos en su almacenamiento y transmisión; así como en el procesamiento manual de esta.
- La disponibilidad de la información necesaria que permite dar visibilidad al centro, está afectada debido a que los procesos de búsqueda son manuales y utilizan datos en diferentes formatos y fuentes.
- No se dispone de una estructura capaz de almacenar la información que se genera en el proceso, y las existentes están dispersas y en diferentes formatos.

A partir de la situación problemática anteriormente expuesta, se plantea como problema de la investigación: ¿Cómo favorecer el proceso de gestión de la información asociada a los servicios que se brindan en el Centro de Innovación y Calidad de la Educación, que contribuya al aumento de los grados de centralidad, disponibilidad, seguridad e integridad de la información?

En correspondencia con el problema planteado, el **objeto de estudio** lo constituye el proceso de gestión de la información asociada a los servicios que se brindan en centros que investigan en el área de las Ciencias de la Educación Superior.

Se define como **campo de acción** el proceso de gestión de la información asociada a los servicios que se brindan en el Centro de Innovación y Calidad de la Educación de la Universidad de las Ciencias Informáticas.

Para dar solución al problema planteado se define como **objetivo general**: Desarrollar un sistema de gestión de información, que contribuya al aumento de los grados de centralidad, disponibilidad, seguridad e

integridad de la información asociada a los servicios que se brindan en el Centro de Innovación y Calidad de la Educación.

Para darle solución al objetivo general se definen las siguientes **tareas de investigación**:

1. Establecimiento de los referentes teórico-metodológicos para la construcción de sistemas informáticos que promuevan los servicios que se brindan en centros que investigan en el área de las Ciencias de la Educación Superior.
2. Caracterización del estado actual del proceso de gestión de la información en el Centro de Innovación y Calidad de la Educación de la Universidad de las Ciencias Informáticas.
3. Fundamentación de las tecnologías, herramientas, metodología y lenguajes necesarios para el desarrollo del producto.
4. Modelación del sistema teniendo en cuenta la metodología seleccionada.
5. Implementación del sistema de gestión de la información para la promoción de los servicios que se brindan en el Centro de Innovación y Calidad de la Educación de la Universidad de las Ciencias Informáticas.
6. Validación del sistema de gestión de la información del Centro de Innovación y Calidad de la Educación de la Universidad de las Ciencias Informáticas a través de las pruebas que se diseñen.
7. Valoración de la propuesta a través de la comprobación de la centralidad, disponibilidad, seguridad e integridad de la información del sistema de gestión de la información del Centro de Innovación y Calidad de la Educación de la Universidad de las Ciencias Informáticas.

Para guiar la investigación se plantea la siguiente **hipótesis**: La introducción de un sistema de gestión de la información en el Centro de Innovación y Calidad de la Educación de la Universidad de las Ciencias Informáticas, contribuirá al aumento de los grados de centralidad, disponibilidad, seguridad e integridad de la información asociada a los servicios que en éste se brindan.

En el desarrollo de la investigación se utilizaron un conjunto de **métodos científicos**. Como parte de los **métodos teóricos** utilizados se encuentran:

- **Método Histórico – lógico**: Se aplicó para la determinación de antecedentes y tendencias actuales del objeto de estudio y el campo de acción. Específicamente para estudiar la forma en que han evolucionado las tecnologías para el desarrollo de sistemas informáticos que promueven los

servicios que se brindan en centros que investigan en el área de las Ciencias de la Educación Superior.

- **Método Análisis – síntesis:** Se aplicó con el objetivo de determinar las generalidades y especificidades en el objeto de estudio y el campo de acción; así como en la fundamentación teórica y elaboración de la propuesta de solución.

Se utilizaron como **métodos empíricos:**

- **Entrevista:** Se aplicó para obtener información sobre los servicios que se brindan en el Centro de Innovación y Calidad de la Educación de la Universidad de las Ciencias Informáticas, así como para la determinación de las posibles funcionalidades y otras características que debía tener la aplicación a construir. Específicamente se utiliza la entrevista no estructurada.
- **Análisis documental:** Se utilizó con el objetivo de obtener información mediante la recolección y selección de documentos relacionados con el tema que se estudia.

Posibles resultados:

1. Documentación ingenieril asociada a los flujos de trabajo: Requisitos, Diseño, Implementación y Pruebas.
2. Sistema de gestión de información para el Centro de Innovación y Calidad de la Educación de la Universidad de las Ciencias Informáticas.
3. Informe resultante del proceso de investigación.

El presente trabajo está compuesto por tres capítulos que se estructuran de la forma descrita a continuación:

Capítulo 1: Fundamentación Teórica: Se desarrolla un estudio del estado del arte para abordar cómo se lleva a cabo la gestión de la información como proceso, asociado a los sistemas informáticos que promueven los servicios que se brindan en centros que investigan en el área de las Ciencias de la Educación Superior. También se analizan diferentes sistemas asociados al campo de acción para determinar la necesidad de una nueva solución. Por último, se fundamenta la selección de la metodología de desarrollo y de las tecnologías a utilizar para el desarrollo de la aplicación.

Capítulo 2: Análisis del Sistema: Se describen las características del sistema a partir de los requisitos funcionales y no funcionales de la aplicación. Se muestra la estructura general del sistema a través de la

relación entre los diversos paquetes que agrupan las funcionalidades identificadas, así como el diagrama de casos de uso del sistema y una descripción textual por cada caso de uso del sistema definido.

Capítulo 3: Diseño del Sistema: Se describe la solución a partir de la definición del patrón arquitectónico a utilizar, la conformación de la vista lógica de la arquitectura, así como los patrones de diseño que se utilizan. Se muestra también una vista del entorno de despliegue del sistema. Por último, se presentan el diagrama de clases persistentes y el diagrama entidad – relación del sistema a construir.

Capítulo 4: Implementación y Pruebas: Se describe el proceso de implementación. Se detallan las pruebas realizadas al sistema para la comprobación del correcto funcionamiento del mismo. Este capítulo concluye con la validación de la contribución lograda a través de la introducción del sistema, en el proceso de gestión de la información del Centro de Innovación y Calidad de la Educación en la Universidad de las Ciencias Informáticas.

Para finalizar se presentan las conclusiones generales, recomendaciones, bibliografía y anexos para un mejor entendimiento de lo expuesto en la investigación.

Capítulo 1. Fundamentación Teórica

Para comprender la siguiente investigación, es necesario describir los aspectos teóricos relacionados con el proceso de gestión de la información asociada a los servicios que se brindan en centros que investigan en el área de las Ciencias de la Educación Superior y particularmente su ejecución en el Centro de Innovación y Calidad de la Educación. De igual forma se especifican de manera sintetizada algunos de los sistemas que existen a nivel nacional e internacional con fines similares. Por último, se caracteriza la metodología de desarrollo escogida y se fundamenta el ambiente de desarrollo que se utiliza para dar respuesta a la propuesta de solución.

1.1 El proceso de gestión de la información. Integridad y disponibilidad de la información

Actualmente, la gestión de la información constituye una necesidad clave de las organizaciones, porque este tipo de tecnología se ve involucrada desde las líneas de producción hasta los niveles que se encargan de la toma de decisiones. De esta forma, los sistemas de procesamiento de información inciden en la manera de trabajar, en la cultura y en la estrategia de muchas de las compañías actuales. El facilitar la comunicación, el procesamiento de datos y la manera de almacenar la información permiten establecer redes, y mediante los procedimientos adecuados, también garantizan descubrir el conocimiento diseminado a lo largo de la organización para convertirlo en material de aprendizaje útil a todos los miembros de la misma.

En el marco de la competencia actual y con los retos que implican una economía global, es necesario que las empresas vean en los sistemas de gestión de la información una herramienta que les permita adquirir una ventaja competitiva sostenible, haga eficientes sus procesos y optimice sus operaciones. Según (Capote, y otros, 2003) la gestión de la información es *«el proceso que se encarga de suministrar los recursos necesarios para la toma de decisiones, así como para mejorar los procesos, productos y servicios de la organización»*. El propio autor expresa que *«la información para que pueda utilizarse y genere ventajas competitivas debe tener tres características básicas: completa, confiable y oportuna»*.

La gestión de la información (GI) es todo lo relacionado con la obtención de la información adecuada, en la forma correcta, para la persona indicada, al costo adecuado, en el tiempo oportuno, en el lugar apropiado, para tomar la acción correcta (Espinosa, 2013). Para (Soto, et al., 2006) la GI se puede definir como *«el*

Capítulo 1. Fundamentación Teórica

conjunto de actividades realizadas con el fin de controlar, almacenar y, posteriormente, recuperar adecuadamente la información producida, recibida o retenida por cualquier organización en el desarrollo de sus actividades». Según (Bartle, 2011) la GI «es el proceso de analizar y utilizar la información que se ha recabado y registrado para permitir a los administradores tomar decisiones documentadas». Este autor plantea que la gestión de información implica:

- **Determinar la información necesaria para la gestión:** durante la planificación, gestión y supervisión del proyecto se genera mucha información. Parte de ella es necesaria para tomar decisiones inmediatas, parte para decisiones posteriores. Un buen sistema de gestión de la información debe, por lo tanto, ayudar a los administradores del proyecto a saber qué información necesitan recabar, para tomar diferentes decisiones en distintos momentos.
- **Obtener y analizar la información para gestionarla:** la información puede conseguirse de informes de técnicos, libros de registro, formularios de los diferentes ejecutantes, reuniones con la comunidad, entrevistas, observación y mapas comunitarios.
- **Registro de la información:** es importante guardar la información para futuras referencias. Puede guardarse en libros de registro locales, informes de progreso, formularios. El principio más importante del registro de informaciones es la facilidad con la que pueden recuperarse.
- **Empleo de la información:** se puede utilizar para solucionar problemas comunitarios, determinar recursos (cantidad y naturaleza), solicitar apoyos y planear futuros proyectos.
- **Divulgación o flujo de información:** para que la información tenga un uso adecuado tiene que compartirse con los demás interesados o usuarios. Esta información puede ayudarles en sus decisiones y también puede ayudar al que la recoge a encontrar significados o usos relacionados con la gestión.

La autora de esta investigación considera necesario tener en cuenta los criterios de (Capote, y otros, 2003) y (Bartle, 2011) sobre la gestión de la información definiéndola como un proceso. Por otra parte, en aras de concretar un concepto más asequible para el contexto de la investigación se asume lo planteado por (Bustelo, y otros, 2001) al definirla como: *«el conjunto de actividades realizadas con el fin de controlar, almacenar y, posteriormente, recuperar adecuadamente la información producida, recibida o retenida por cualquier organización en el desarrollo de sus actividades».*

La finalidad de la gestión de la información es ofrecer mecanismos que permitan a la organización adquirir, producir y transmitir, al menor coste posible, datos e informaciones con una calidad, exactitud y actualidad suficientes para servir a los objetivos de la organización. En términos perfectamente entendibles sería conseguir la información adecuada, para la persona que la necesita, en el momento que la necesita, al mejor precio posible para tomar la mejor de las decisiones.

La palabra “integridad”, según (DRAE), significa: «cualidad de íntegro». De igual forma, (DRAE) plantea que “íntegro” refiere «que no carece de ninguna de sus partes». Para (McGraw-Hill, 2001) la integridad representa: *«completitud y corrección de los datos almacenados en una computadora, especialmente después de que ésta ha sido manipulada de alguna forma»*. Los propios autores definen además el concepto de integridad de datos siendo *«la precisión de los datos y su conformidad al valor esperado, especialmente después de ser transmitidos o procesados»*. A su vez, (IEEE Std 610.12, 1990) define la integridad como *«el grado en que un sistema o componente impide el acceso no autorizado a, o la modificación de los programas informáticos o los datos»*.

Para el marco de la investigación la autora de la misma propone que se vea la integridad de la información como el grado de completitud, precisión y conformidad al valor esperado de la información después que esta ha sido almacenada de alguna forma y es requerida para su uso. Para definir lo que representa la disponibilidad de la información, la propia autora coincide con lo planteado en (IEEE Std 610.12, 1990), donde es definida como *«el grado en que un sistema o componente está operativo y accesible cuando se requiere para su uso. A menudo se expresa como una probabilidad.»*

1.2 Caracterización del proceso de gestión de la información en el Centro de Innovación y Calidad de la Educación

Hoy día, el proceso de gestión de la información en el CICE de la UCI (PGICICE-UCI) ocurre bajo la estructura administrativa del propio centro. El objetivo de gestionar la información asociada a los servicios que se brindan en el CICE es controlarla, almacenarla y obtenerla adecuadamente para tomar decisiones. Para cumplir este objetivo es necesario garantizar la integridad, centralidad, seguridad y disponibilidad de la información, después de haber sido almacenada. Sin embargo, se constatan deficiencias en el PGICICE-UCI, las cuales se expusieron en la situación problemática descrita en la introducción de esta memoria.

Capítulo 1. Fundamentación Teórica

Se constata un procesamiento de la información de Ciencia, Tecnología e Innovación (CTI) de forma manual. Una de las informaciones que se gestiona en CTI, es la relacionada con los eventos y revistas. Para el control de las antes mencionadas se confecciona un Excel que tiene una pestaña donde se registra el título de la revista, la editorial, las temáticas, el país, la frecuencia de publicación, el idioma, el nivel del MES al que pertenece, bases de datos en la que se encuentra indexada e ISSN. Tiene otra pestaña asociada a los eventos donde se registra el nombre, la fecha en que se realizará, el tipo de evento, el país, lugar, el programa, la Url. Además, se le envía a cada trabajador una plantilla Word para que llenen los resultados que se obtienen en el período que se solicita por cada uno de los indicadores de Ciencia y Técnica y se envían por correo las evidencias. Existe una persona del grupo que se encarga de compilar toda esta información en un fichero Excel, procesarla y realizar las gráficas correspondientes.

La necesidad de preparar desde el punto de vista pedagógico a los profesores de la Universidad de Ciencias Informáticas se ve reflejada en una de las áreas del CICE, la de formación pedagógica. Tiene como misión instrumentar acciones encaminadas a la preparación pedagógica y didáctico-metodológica del claustro de las diferentes facultades. Desde este punto de vista se acometen acciones de preparación a los Recién Graduados en Adiestramiento (RGA), las que favorecen su preparación para impartir las asignaturas del ciclo específico de la carrera de Ingeniería en Informática.

A estos profesores se les imparte el Curso Básico de Pedagogía y Didáctica. También se trabaja en la profundización y actualización de los conocimientos acerca de los temas relacionados con la Pedagogía y la Didáctica de la Educación Superior para los que se desarrollan: cursos independientes, entrenamientos, diplomados, maestrías, doctorados. Se ha demostrado que existe un bajo grado de accesibilidad a la información para esta área de significativa importancia, provocado por no tener espacios propios que le permitan al centro divulgar sus principales tareas, resultados e información de interés como la misión, visión, los trabajadores, su estructura de trabajo; para lo cual se deben auxiliar del correo electrónico, sitios web de las facultades, discusiones en reuniones y talleres que se organizan por el centro, plegables.

La caracterización realizada evidencia insuficiencias en los grados deseados de centralidad, integridad y disponibilidad de la información en el PGICICE-UCI. Lo expuesto anteriormente, constata la existencia del problema científico y se reafirma el objetivo propuesto.

1.3 Análisis de las soluciones existentes

Portal Web del Centro de Referencia para la Educación de Avanzada (CREA):

El Crea es uno de los Centros de Investigación del Instituto Superior Politécnico "José Antonio Echeverría (CUJAE), cuya misión es la de "Desarrollar y potenciar los estudios avanzados y la formación de recursos humanos en el campo de las Ciencias Pedagógicas y Ciencias de la Educación, aplicados fundamentalmente a la enseñanza de las Ciencias Técnicas, la Ingeniería y la Arquitectura, haciendo énfasis en la aplicación de las tecnologías de la información y las comunicaciones en la Educación Superior y en su proceso de Universalización". De su misión se desprenden dos líneas fundamentales de trabajo: La formación pedagógica de profesores universitarios y la aplicación de las tecnologías de la información y las comunicaciones en el proceso docente. El trabajo del centro se proyecta hacia todas las carreras y centros de investigación de la Cujae y se apoya en el trabajo de colaboradores, tanto del propio Instituto como de otros Centros nacionales y extranjeros. Su labor se centra en las siguientes Áreas de Resultados Clave: Superación de Posgrado, Ciencia y Técnica e Informatización".

El centro cuenta con un portal web que tiene un menú con cuatro secciones. La primera es la página principal donde se encuentran varios apartados, entre los que se encuentran el buscador, varios vínculos asociados a laboratorios virtuales, a temáticas relacionadas con las TIC en el aprendizaje de las Ciencias Técnicas. En el pie de página se encuentran las mismas secciones y vínculos al portal de la Cujae para contactarlos y ver sus créditos.

Al desplegar la sección "Nuestro Centro" se observan varias opciones que describen al mismo como son: ¿Quiénes somos?, donde se explica el objetivo del centro, su misión, las personas que lo conforman y Principales Resultados, donde se exponen los diferentes resultados y los proyectos que se han desarrollado. En la opción Trabajadores, se muestran los miembros del centro con su foto, nombre, área a la que pertenecen y su información de contacto. Por su parte en Departamento de Producción, se describe el área de producción; en Departamento de Informatización, se muestran los servicios que brinda y garantiza este departamento además de las aplicaciones que desarrollan y dan mantenimiento a la Teleformación en la Cujae. Por último, en el Centro de Documentación, se observa una breve descripción del mismo.

La sección de Oferta Académica, se encuentra dividida en dos partes: Formación Pedagógica Básica y Posgrado. En la primera se muestran los cursos de Pedagogía que reciben los estudiantes. La segunda

Capítulo 1. Fundamentación Teórica

tienes varias opciones como: Cursos, Diplomados, Maestrías, Doctorados, de las cuales se publican los diferentes tipos de posgrados que se imparten y el lugar en que se realizan. Se debe resaltar que los doctorados tienen asociados archivos adjuntos en formato .doc y .pdf que se pueden descargar. La última sección “Investigaciones”, hace referencia a los proyectos nacionales y a los proyectos y redes internacionales que tienen y permite descargar el documento que se encuentra en formato .pdf.

Portal Web del Centro de Estudios para el Perfeccionamiento de la Educación Superior (CEPES):

El CEPES es un centro de estudios de la Universidad de La Habana cuya misión es la de contribuir, desde una perspectiva innovadora, a la satisfacción de las demandas de posgrado de directivos, profesores e investigadores, así como del desarrollo científico – tecnológico en el área de las Ciencias de la Educación Superior y del fomento de la cultura integral de la población, en correspondencia con las exigencias de la universidad cubana actual. Para cumplir con dichos propósitos, forma profesionales competentes con un alto nivel de calificación a través de diferentes modalidades de superación académica y profesional, desarrolla continuamente la investigación científica mediante la generación, asimilación, adaptación, difusión y transferencia de conocimientos y proyecta su trabajo extensionista con acciones orientadas a la promoción cultural de sus resultados en la sociedad, lo que contribuye a elevar la calidad de la labor educativa y el trabajo político ideológico y al perfeccionamiento de la gestión de los procesos sustantivos de la educación superior.

El portal de este centro cuenta con 13 secciones: Quiénes somos, Noticias, Órganos Asesores, Grupos de Trabajo, Posgrado, Investigaciones, Cátedra Unesco, Directorio, Enlaces, Próximos Eventos, Revista Cubana, Tribunal Permanente y Redes Académicas. En la primera de estas se describe el centro, se muestra su visión, el claustro que lo compone, los reconocimientos que se le han otorgado, los grupos de trabajo. En Noticias se muestra información asociada a los acontecimientos de gran relevancia que guardan estrecha relación con el trabajo que se realiza en el centro. En la sección de Órganos Asesores se presenta el consejo de dirección, consejo científico, el organigrama y el claustro en por ciento por categorías docentes.

En Grupos de Trabajo, se realiza una breve descripción de cada uno de ellos. En Posgrado se muestra un listado de las tesis defendidas en cada edición de los programas de maestrías, así como gráficos asociados al programa de maestría. Se incluyen también los programas de doctorado impartidos y la cantidad de graduados de actividades en Posgrado (Cursos, Entrenamientos, Diplomados, Maestría, Doctorado). En la sección de Investigaciones, se encuentran las líneas de investigación, los núcleos básicos de investigación

Capítulo 1. Fundamentación Teórica

por áreas, los proyectos en ejecución, los premios recibidos en los últimos años, los libros publicados con participación de autores del centro en los últimos años, artículos publicados y documentos asociados al balance de investigaciones de diferentes años.

Portal Web de la Asociación Universitaria Iberoamericana de Posgrado (AUIP):

Es un organismo internacional no gubernamental reconocido por la UNESCO, que se dedica al fomento de los estudios de posgrado y doctorado en Iberoamérica. La AUIP presta servicios de información y divulgación sobre los posgrados que se ofrecen. Colabora en procesos de evaluación interna y externa, acreditación y armonización curricular de esa oferta académica. Facilita la movilidad e intercambio de profesores y estudiantes. Incentiva el trabajo académico a través de redes de centros de excelencia en diversos campos del conocimiento. Auspicia eventos académicos y científicos que están claramente relacionados con la formación avanzada y organiza cursos itinerantes internacionales en temas de interés para profesores y directores de programas de posgrado y doctorado.

El portal de la AUIP está compuesto por un menú de 5 secciones. En la página principal, que es la primera sección, tiene un carrusel donde promocionan las becas que ofertan recientemente. Incluye una sección de actividades donde se encuentran las becas que se están ofertando, las cuales se clasifican en: Becas de Movilidad, Becas de Máster y Becas de Doctorado. Se muestran los cursos de posgrado iberoamericanos de diferentes años; los premios AUIP a la calidad del posgrado en Iberoamérica, una guía de evaluación de la sexta edición y las pautas para la evaluación externa en formato .pdf. Además, se exponen las misiones técnicas, las redes de investigación, las bases y criterios para obtener el premio “Ignacio Ellacuría” a la cooperación del posgrado en Iberoamérica y las diferentes publicaciones que han tenido. De igual forma existe una sección: “Más Información”, donde se publican las becas y premios de Universidades asociadas. En la opción de Oferta de Posgrado se puede acceder directamente a las páginas relacionadas con la oferta de posgrado y doctorado de cada una de las universidades asociadas y se ofertan otras becas de posgrado. También permite la suscripción en el boletín informativo, presenta la documentación asociada a esta asociación con la oportunidad de descargarla y enlaces a diferentes entidades. La segunda sección que es la de “Presentación”, está conformada por una descripción de la AUIP, sus fines y objetivos, sus estatutos, los informes de gestión y los programas de acción que realizan. En la tercera sección, llamada “Instituciones Asociadas”, se muestran las instituciones que conforman la AUIP repartidas en diferentes países entre los que se encuentran: Argentina, Bolivia, Cuba, Venezuela, Ecuador, y otros. La sección de “Órganos de

Capítulo 1. Fundamentación Teórica

Gobierno” se encuentra dividida en: Comisión Ejecutiva y Dirección General. Y en la sección “Afiliación a la AUIP” se muestran los pasos del proceso para solicitar afiliarse a esta asociación.

Portal Web del Centro de Estudios Sanitarios (CPES CESA BHIP):

El Centro de Estudios Sanitarios Homologado por el Departamento de Educación del Gobierno Vasco, para la impartición de Ciclos de Grado Medio y Superior, se dedica a la formación de profesionales en diversas especialidades sanitarias. CPES CESA BHIP es un Instituto de Formación Profesional Superior. En él se imparte formación profesional (ciclos formativos de grado medio y superior), formación continua (cursos para trabajadores y desempleados) y formación ocupacional.

Este centro cuenta con un portal que tiene 6 secciones: la primera de ellas es la página principal donde se encuentra una pequeña descripción del centro, entidades que colaboran, noticias y cursos destacados. En la sección “El Centro” se visualiza una descripción más amplia del centro y una galería de imágenes de sus instalaciones. La sección “Cursos” es donde se ofertan diferentes cursos clasificados en: Grado Superior, Grado Medio, Formación Ocupacional, Formación Continua y Cursos para Empresas. Tiene una sección para las noticias, otra para que puedan ser contactados, que muestra datos como lugar, teléfono y dirección de correo electrónico. Por último, se muestra la sección “Centro de Aprendizaje Virtual”, que contiene un buscador para los cursos y permite que las personas puedan recibir estos cursos de forma online.

Portal Web del Centro de Innovación y Calidad de la Educación (CICE), versión 1.0:

El CICE es un centro de estudios de la Universidad de las Ciencias Informáticas, que contribuye a elevar la calidad de los procesos fundamentales de la institución. El portal web del centro tiene una sección “Bienvenido” donde se muestra la misión, visión e ideas rectoras del mismo. Además, tiene otra sección llamada “CICE_Academia”, que está dividida en varias secciones como son: ¿Qué es el CICE?, “Calidad del Posgrado”, “Investigación”, “Innovación Educativa”, “Colaboración Interinstitucional”, “Gestión de la Información”, “Gestión Calidad”, “Gestión Eventos” y “APC Filial UCI”.

Al desplegar la primera de las secciones, se muestran datos como los valores, objetivos, la estructura y el equipo de trabajo del centro. En la segunda, se publican conferencias y talleres, cursos de posgrado, diplomados, maestrías, doctorados que se ofrecen y un canal de cultura pedagógica. La tercera sección contiene la línea de investigación del centro. En la cuarta contiene la información asociada a los proyectos de innovación educativa como: las convocatorias, las plantillas, los proyectos activos, culminados e

interrumpidos. En la sección “Colaboración Interinstitucional”, que es de tipo nacional e internacional se muestran las instituciones que colaboran con el centro y sus proyectos. La sección “Gestión de la Información” contiene un centro de información de diferentes temáticas. “Gestión Calidad” abarca un conjunto de documentos relacionados con la calidad en la Educación Superior. En “Gestión Eventos” se muestran los eventos a realizarse en la UCI, en Cuba y en el extranjero.

En la última, “APC Filial UCI” se encuentra la estructura y talleres asociados al aprendizaje del uso de las tecnologías en la educación. Existe otra sección general llamada “CICE_Comunidad” que contiene las noticias, artículos, preguntas frecuentes, los vínculos a revistas y otras publicaciones, foro de discusión, galería de imágenes, un buscador. Posee un calendario de eventos, una sección del usuario con su perfil, avatar, los usuarios en línea, tiene en el menú un vínculo a las tesis de la biblioteca y enlaces a los sitios de la Intranet e Inter-Nos de la universidad. Este portal se encuentra muy cargado de información y el esbozo no se encuentra acorde a las exigencias de diseño web actuales.

Conclusiones de los sistemas estudiados

Se concluye que no es posible la adopción de ninguno de los sistemas estudiados ya que no presentan de manera integrada todas las funcionalidades que se requieren para satisfacer las necesidades del centro de estudio alusivo. Todos responden a un negocio específico según los procesos que realizan y la estructura propia de la organización y fueron desarrollados con tecnologías, que en la actualidad son obsoletas.

No obstante, se debe destacar que el análisis de los sistemas anteriormente descritos, sirvió para identificar funcionalidades básicas que pueden implementarse en la propuesta de solución entre las que se encuentran las asociadas a la información general del centro como la visión, misión, objetivos, grupos de trabajo y trabajadores del mismo. De igual forma se tiene en cuenta la oferta académica en el pregrado y en el posgrado que incluyen los cursos de posgrado, maestrías, doctorados, diplomados; las investigaciones donde se exponen los proyectos; promoción de eventos; galería de imágenes; publicaciones en eventos y documentos de interés del centro.

1.4 Tendencias y tecnologías actuales

En este apartado se definen las tecnologías que permiten organizar y preparar el proceso de desarrollo del sistema que se desea construir. Se selecciona la metodología, lenguaje de modelado, herramientas CASE, lenguajes de programación, servidores web y sistemas gestores de base de datos. Se justifica dicha

selección y se realiza una descripción de los mismos para el conocimiento de sus características, funcionalidades y ventajas.

1.4.1 Metodología de desarrollo de software Proceso Unificado Abierto (OpenUP)

OpenUP es un proceso de desarrollo iterativo del software que es mínimo, completo y extensible. Es mínimo porque solamente el contenido fundamental es incluido; es completo debido a que puede ser manifestado como todo el proceso para construir un sistema; extensible ya que puede ser utilizado como fundamento sobre el cual el contenido de proceso se pueda agregar o adaptar según lo necesitado (Wesley, 2004).

OpenUP es un proceso ágil, iterativo e incremental basado en *Rational Unified Process* (RUP), que contiene el conjunto mínimo de prácticas que ayudan a los equipos de desarrollo de software a realizar de manera eficiente un producto con alta calidad. Metodología apropiada para proyectos pequeños y es aplicable a un conjunto extenso de plataformas y aplicaciones de desarrollo. Dirigido por casos de uso y centrado en la arquitectura durante el ciclo de vida del software. Utiliza como lenguaje de modelado el UML (Lenguaje Unificado de Modelado).

Las iteraciones realizadas en este proceso se distribuyen a través de cuatro fases:

Inicio: En esta fase se toman en cuenta todas las necesidades de cada participante del proyecto. Se definen para el proyecto: el ámbito, los límites, el criterio de aceptación, los casos de uso críticos, una estimación inicial del costo y un boceto de la planificación.

Elaboración: En esta fase se realizan tareas de análisis del dominio y definición de la arquitectura del sistema. Se debe elaborar un plan de proyecto, estableciendo unos requisitos y una arquitectura estables. También se especifican en detalle el proceso de desarrollo, las herramientas, la infraestructura a utilizar y el entorno de desarrollo. Al finalizar se debe tener una definición clara y precisa de los casos de uso, los actores, la arquitectura del sistema y un prototipo ejecutable de la misma.

Construcción: Todos los componentes y funcionalidades del sistema que falten por implementar son realizados, probados e integrados en esta fase. Los resultados obtenidos en forma de incrementos ejecutables deben ser desarrollados de la forma más rápida posible sin dejar de lado la calidad de lo desarrollado.

Transición: Esta fase corresponde a la introducción del producto en la comunidad de usuarios, cuando el producto está lo suficientemente maduro. La fase de la transición consta de las subfases de pruebas de versiones beta, pilotaje y capacitación de los usuarios finales y de los encargados del mantenimiento del sistema. En función de la respuesta obtenida por los usuarios puede ser necesario realizar cambios en las entregas finales o implementar alguna funcionalidad más.

Se decidió seleccionar esta metodología para el proceso de desarrollo de la aplicación porque es centrada al cliente y con iteraciones cortas. Permite descubrir errores tempranos a través de ciclos iterativos, lo que evita la elaboración de documentos, diagramas e iteraciones innecesarias. Además, es apropiada para proyectos pequeños y de bajos recursos. También posibilita disminuir las probabilidades de fracaso en los proyectos pequeños e incrementar las probabilidades de éxito.

1.4.2 Lenguaje unificado de modelado (UML 2.0)

El Lenguaje Unificado de Modelado (en inglés Unified Model Language), según (Jacobson, y otros, 2000) *«permite modelar analizar y diseñar sistemas orientados a objetos y es el lenguaje de modelado de sistemas de software más conocido y utilizado en la actualidad»*. Sirve para especificar, visualizar y documentar esquemas de sistemas de software orientado a objetos.

Un modelo UML está compuesto por elementos, relaciones y diagramas. Permite la modelación del ciclo completo de desarrollo de software. Define trece tipos de diagramas, divididos en tres categorías: Seis tipos de diagramas representan la estructura de aplicación estática; tres representan tipos generales de comportamiento; y cuatro representan diferentes aspectos de las interacciones.

1.4.3 Herramienta Case Visual Paradigm for UML 8.0

Visual Paradigm es una herramienta de Ingeniería de Software Asistida por Computadora (CASE, por sus siglas en inglés) para el modelado con UML. Soporta el ciclo de vida completo del desarrollo de software: análisis y diseño orientado a objetos, construcción, pruebas y despliegue. Además, facilita a los ingenieros de software diseñar, integrar y modelar visualmente los distintos diagramas que se generan a lo largo del desarrollo del software. Presenta un generador de código que soporta más de 10 lenguajes (Java, C++, PHP.) y proporciona la ingeniería inversa (Paradigm, 2013). Se integra con varios entornos de desarrollo como VisualStudio, NetBeans, Eclipse. Esta herramienta es eficiente, fácil de usar, con soporte

multiplataforma. Proporciona excelentes facilidades de interoperabilidad con otras aplicaciones. Tiene licencia dual, gratuita y comercial.

1.4.4 Lenguajes empleados en la programación

Los lenguajes de programación representan un idioma artificial diseñado para expresar procesos que pueden ser llevados a cabo por máquinas como las computadoras. Pueden utilizarse para la creación de programas que controlen el comportamiento físico y lógico de una máquina, para expresar algoritmos con precisión, o como modo de comunicación. Están formados por un conjunto de símbolos y reglas sintácticas y semánticas que definen su estructura. La selección de cada lenguaje se realiza de acuerdo a las características y al tipo de software a desarrollar (Meyer, 2006).

1.4.4.1 PHP 5.6

PHP es el acrónimo de *Hypertext Preprocessor* (Preprocesador de Hipertexto). Se trata de un lenguaje interpretado de alto nivel embebido en páginas HTML y ejecutado en el servidor. Está muy orientado al desarrollo de aplicaciones web y permite insertar contenidos dinámicos en las páginas. El principal objetivo de PHP 5 ha sido mejorar los mecanismos de programación orientada a objetos. Un paso necesario para lograr que PHP sea un lenguaje apto para todo tipo de aplicaciones y entornos. Es uno de los lenguajes más utilizados en la web para la realización de páginas avanzadas para todo tipo de entornos. Se caracteriza por ser Multiplataforma, se ejecuta del lado del servidor, con licencia GPL o de software libre, presenta una sintaxis cómoda orientado a objetos completamente, presenta un gran número de extensiones o bibliotecas que amplían sus funcionalidades (Merley, 2012). La característica más potente y destacable es su soporte para una gran cantidad de bases de datos. Es un lenguaje fácil de asimilar y utilizar lo que permite reducir el tiempo empleado en estas funciones. Posee una comunidad de desarrolladores amplia y activa de manera que la mayor parte de los contratiempos en la implementación pueden ser solucionados con rapidez. Existe un número elevado de marcos de trabajo que toman como base este lenguaje entre los que destaca Symfony2.

1.4.4.2 HTML 5

De las siglas de (*Hyper Text Markup Language*), es un lenguaje de marcado usado para estructurar y presentar el contenido para la web. HTML 5 es la quinta revisión del estándar que fue creado en 1990. Con el uso de HTML 5, se puede reducir la dependencia de los *plugins* que tenemos que tener instalados para

poder ver una determinada web y se amplía el horizonte del desarrollo de aplicaciones que pueden ser usadas en una multiplicidad de dispositivos. Permite que los usuarios accedan a sitios web sin estar conectados a internet. Cuenta con nuevas etiquetas que ayudan a los desarrolladores a mejorar la estructura de las páginas como: `<header>` para las cabeceras, `<article>` para definir un artículo, un comentario de usuario o una publicación independiente dentro del sitio, `<section>` define todo tipo de secciones dentro de un documento, `<footer>` para el pie de página. Este es soportado por las versiones actuales de los navegadores Mozilla Firefox, Chrome, Chromiun, Safari e Internet Explorer.

1.4.4.3 CCS 3

CSS (*Cascading Style Sheets*): es un lenguaje de hojas de estilos en cascada creados para controlar la presentación de los documentos electrónicos definidos con HTML y XHTML. CSS 3 está dividido en varios documentos llamados módulos. Cada módulo cuenta con nuevas capacidades, sin afectar la compatibilidad de la versión estable anterior. Al hablar de módulos, se pueden nombrar más de cincuenta, sin embargo, cuatro de ellos han sido publicados como recomendaciones formales, y se componen de lo siguiente:

- *Media Queries* (publicado en 2012)
- *Namespaces* (publicado en 2011)
- *Selectors Level 3* (publicado en 2011)
- *Color* (publicado en 2011).

Estas hojas de estilos permiten crear diseños atractivos y personalizados de acuerdo a los requisitos no funcionales de apariencia o interfaz externa.

1.4.4.4 JavaScript v1.2

JavaScript es un lenguaje de programación interpretado, se utiliza principalmente en su forma del lado del cliente, implementado como parte de un navegador web permitiendo mejoras en la interfaz de usuario y páginas web dinámicas. Los programas escritos con *JavaScript* se pueden probar directamente en cualquier navegador sin necesidad de procesos intermedios. Es utilizado para manejar los elementos gráficos del sistema con el fin de lograr una interfaz de usuario atractiva, dinámica y de fácil utilización para el usuario final.

1.4.5 Marco de trabajo

1.4.5.1 *Symfony v2.8*

Symfony es un completo *framework* diseñado para optimizar, gracias a sus características, el desarrollo de las aplicaciones web. Está basado en el patrón Modelo-Vista-Controlador. Separa la lógica de negocio, la lógica de servidor y la presentación de la aplicación web. Tiene varias herramientas y clases que permiten reducir el tiempo de desarrollo de una aplicación web compleja. Symfony es compatible con la mayoría de gestores de bases de datos, como MySQL, PostgreSQL, Oracle y Microsoft SQL Server. Se puede ejecutar tanto en plataformas (Unix, Linux.) como en plataformas *Windows*. Posee una comunidad de desarrolladores amplia y activa de manera que la mayor parte de los contratiempos en la implementación pueden ser solucionados con rapidez y efectividad. Brinda una arquitectura sólida y efectiva que asegura una solución escalable de manera que puede aumentar en funcionalidades sin afectar o afectando de manera poco significativa las funcionalidades que se brindan en versiones anteriores. Presenta un amplio repositorio de componentes libres, desarrollados por la comunidad que permiten agilizar el desarrollo de manera significativa.

1.4.5.2 *Bootstrap v3.1*

Bootstrap es un *framework* o conjunto de herramientas de software libre para diseño de sitios y aplicaciones web. Contiene plantillas de diseño con tipografía, formularios, botones, cuadros, menús de navegación y otros elementos de diseño basado en HTML y CSS, así como, extensiones de JavaScript opcionales adicionales.

Características de Bootstrap:

- Compatible con la mayoría de los navegadores *web*.
- Se integra perfectamente con las principales librerías JavaScript, por ejemplo, JQuery.
- Ofrece un diseño sólido a través del uso de estándares como CSS3/HTML5.
- Permite crear interfaces que se adapten a los diferentes navegadores, tanto de escritorio como *tablets* y móviles a distintas escalas y resoluciones.

La gran popularidad alcanzada por este *framework* permite encontrar una amplia gama de componentes visuales que agilizan el tiempo de desarrollo de las interfaces de usuario.

1.4.5.3 JQuery v1.10.2

JQuery es una biblioteca de *JavaScript* que permite simplificar la manera de interactuar con los documentos HTML, manipular el árbol DOM, manejar eventos, desarrollar animaciones y agregar interacción con la técnica *AJAX* a páginas web. *JQuery* es la biblioteca de *JavaScript* más utilizada. Ofrece una serie de funcionalidades basadas en *JavaScript* que de otra manera requerirían de mucho más código, es decir, con las funciones propias de esta biblioteca se logran grandes resultados en menos tiempo y espacio.

Características de JQuery:

- Manipulación de la hoja de estilos CSS.
- Efectos y animaciones.
- Animaciones personalizadas.
- Soporta extensiones.
- Utilidades varias como obtener información del navegador, operar con objetos y vectores, funciones para rutinas comunes.
- Compatible con los navegadores *Mozilla Firefox*, *Internet Explorer*, *Safari*, *Opera* y *Google Chrome*.

1.4.6 Entorno de Desarrollo Integrado (IDE) PHP Storm 7.4

A partir de la selección del lenguaje de programación, es necesario definir un entorno de desarrollo integrado (IDE, por sus siglas en inglés) que permita desarrollar aplicaciones que utilicen el lenguaje PHP. Estudios de (Okeke Oluchukwu, 2015) arrojaron los resultados que se muestran en la (Tabla 1).

Tabla 1 Lista de IDE más mencionados. [tomado de (Okeke Oluchukwu, 2015)]

IDE	Mencionado
PHPStorm	81 veces
Eclipse PDT	33 veces
NetBeans	29 veces
Zend Studio	17 veces

Capítulo 1. Fundamentación Teórica

Komodo IDE	9 veces
phpDesigner	1 vez

Basándose en los indicadores generales, la tendencia del mundo actual y los resultados arrojados en las Figuras (1 y 2), se selecciona *PHPStorm* como IDE de desarrollo. Este IDE fue desarrollado por una división perteneciente a Google y provee un amplio espectro de funcionalidades orientadas a facilitar el trabajo del desarrollador. Se integra con sistemas de bases de datos, sistemas de control de versiones y presenta un excelente completamiento de código. Cuenta con varios *plugins* que aumentan las funcionalidades del IDE, de acuerdo a las necesidades del proyecto.

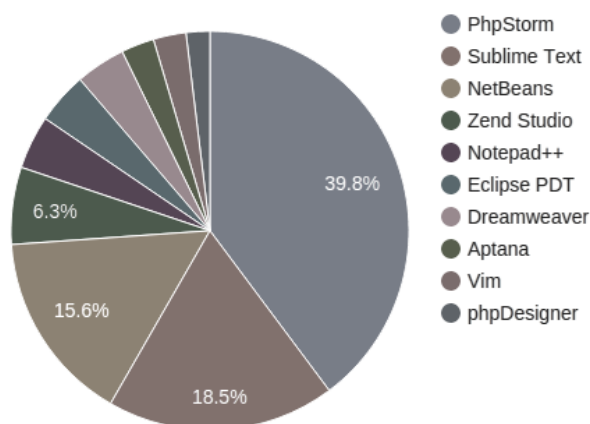


Figura 1 Uso Personal de IDE y editores de PHP. [tomado de (Okeke Oluchukwu, 2015)]

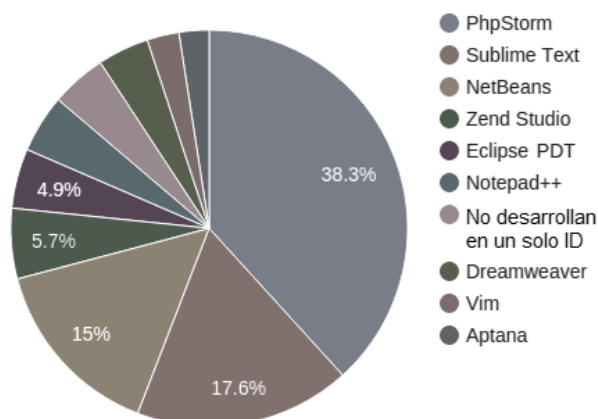


Figura 2 Uso en Negocios de IDE y editores de PHP. [tomado de (Okeke Oluchukwu, 2015)]

Según (Gadja, 2007) «*PHPStorm es un IDE diseñado específicamente para desarrollar aplicaciones de internet escritas en PHP. Las principales características se dividen en las siguientes categorías*».

- Navegación.
- Sincronización de ficheros.
- Depuración.
- Edición y refactorización de código.

El propio autor aborda también que «*las aplicaciones modernas, consisten en muchos archivos, librerías, módulos y clases almacenados en diferentes carpetas. PHPStorm, ayuda a la navegación entre ellas. Los archivos abiertos se encuentran distribuidos en pestañas y es posible navegar entre ellos con el mouse y los accesos del teclado*».

1.4.7 Servidor Web Apache 2.4.4

Apache es un servidor web HTTP de código abierto, para plataformas Unix (BSD, GNU/Linux.), Microsoft Windows, Macintosh y otras, que implementa el protocolo HTTP/1.1 y la noción de sitio virtual. Una de sus características fundamentales es la elevada seguridad que le brinda a sus clientes y es uno de los servidores HTTP más utilizado en la mayoría de los sitios web en el mundo. Se caracteriza por su gran escalabilidad, seguridad y rendimiento. Permite alojar desarrollos implementados en lenguaje PHP e interpretar su código fuente de manera que los usuarios puedan acceder a los contenidos mediante un navegador web.

1.4.8 Sistema gestor de base de datos PostgreSQL v9.4.5

Según (Cobo, 2007) «un SGBD permite el almacenamiento, manipulación y consulta de datos pertenecientes a una base de datos organizada en uno o varios ficheros. En el modelo más extendido (base de datos relacional), la base de datos consiste, de cara al usuario, en un conjunto de tablas entre las que se establecen relaciones». En la propuesta de solución no existe la necesidad de la selección de un SGBD específico. Symfony, está configurado para soportar tanto MySQL como SQLite, PostgreSQL, Oracle, IBM DB2 y SQLServer.

Además, trae incorporado ORM Doctrine que según (Dunglas, 2013) la «biblioteca Doctrine's ORM permite una fácil persistencia y recuperación de gráfico de objetos PHP, sin necesidad de escribir consultas SQL a mano. También proporciona un lenguaje potente orientado a objetos similar a SQL, una herramienta de generación de esquema de base de datos, un sistema de eventos y mucho más». Por otra parte, entre los plugins del IDE seleccionado se encuentra una herramienta para base de datos y SQL que permite la conexión y el manejo de los SGBD como DB2, Derby, H2, HSQLDB, MySQL, Oracle, PostgreSQL, SQLServer, SQLite y Sysbase (Vink, 2012). En el caso de la presente investigación se selecciona PostgreSQL por las ventajas y características que tiene, las cuales se sintetizan a continuación.

PostgreSQL es un sistema de gestión de bases de datos relacional orientado a objetos y libre, publicado bajo la licencia Distribución de Software Berkeley (Berkeley Software Distribution) BSD y con su código fuente disponible libremente. Es el sistema de gestión de bases de datos de código abierto más potente del mercado. Algunas de sus principales características son:

- Soporta distintos tipos de datos: además del soporte para los tipos base, también soporta datos de tipo fecha, monetarios, elementos gráficos, datos sobre redes (MAC, IP), cadenas de bits. También permite la creación de tipos propios.
- Incluye herencia entre tablas, por lo que a este gestor de bases de datos se le incluye entre los gestores objeto-relacionales.
- Copias de seguridad en caliente (Online/hot backups)
- Regionalización por columna
- Múltiples métodos de autenticación
- Acceso encriptado vía SSL
- Completa documentación

- Licencia Berkeley Software Distribution (BSD)
- Disponible para Linux y UNIX en todas sus variantes (AIX, BSD, HP-UX, SGI IRIX, Mac OS X, Solaris, Tru64) y Windows 32/64bit.

La versión utilizada agrega nuevas características que mejoran la flexibilidad, escalabilidad y rendimiento de PostgreSQL para diferentes tipos de usuarios de bases de datos. Incluye mejoras al soporte para JSON, replicación y rendimiento de los índices.

Conclusiones parciales

- La caracterización de la integridad de la información en el proceso de gestión de información asociada a los servicios que se brindan en el Centro de Innovación y Calidad de la Educación, reflejan un bajo nivel de completitud, precisión, corrección semántica, corrección sintáctica y accesibilidad de la información.
- La investigación realizada sobre las diferentes soluciones existentes permitió concluir que ninguna se ajusta a las necesidades expuestas en la investigación, ya que las funcionalidades que poseen se ajustan a un negocio específico, no generalizables a otros entornos, desarrolladas con tecnologías obsoletas, no configuradas para su reutilización y no cumplen con las tendencias y estándares del desarrollo web actuales.
- Las herramientas y tecnologías escogidas para el desarrollo de la aplicación, fueron determinadas por las características de ésta, entre las que se encuentran el marco de trabajo Symfony2, el gestor de base de datos PostgreSQL, Visual Paradigm para la generación de artefactos y la metodología OpenUP para documentar el proceso de desarrollo.

Capítulo 2. Análisis del sistema

En este capítulo se exponen las principales características y cualidades del sistema a desarrollar mediante los requisitos funcionales y no funcionales. Por último, se realiza el diagrama de casos de uso del sistema, así como la descripción de sus actores y casos de uso.

2.1 Requisitos de Software

Una etapa inicial y muy importante dentro del proceso de la Ingeniería de Software, es la Ingeniería de Requisitos. En ésta se realiza el proceso de descubrir, analizar, escribir y verificar los servicios y restricciones, del sistema de software que se desea producir; este proceso se realiza mediante la obtención, el análisis, la especificación, la validación y la administración de los requisitos del software. Los requisitos de software son las necesidades de los clientes, los servicios que los usuarios desean que proporcione el sistema de desarrollo y las restricciones en las que debe operar (Jacobson, y otros, 2000).

Los requisitos de software son las necesidades de los clientes, los servicios que los usuarios desean que proporcione el sistema de desarrollo y las restricciones en las que debe operar (Jacobson, y otros, 2000).

2.1.1 Especificación de Requisitos Funcionales

Los requisitos funcionales son condiciones o capacidades que el sistema debe. A partir de las entrevistas realizadas y la aplicación de otras técnicas de recopilación de información se definieron los siguientes requisitos:

RF1. Adicionar usuario	RF70. Adicionar tipo de evento	RF138. Cambiar contraseña
RF2. Modificar datos de usuario	RF71. Modificar datos de tipo de evento	RF139. Registrar usuario
RF3. Visualizar listado de usuarios	RF72. Visualizar listado de tipos de eventos	RF140. Autenticar usuario
RF4. Mostrar datos de usuario	RF73. Eliminar tipo de evento	RF141. Reestablecer contraseña
RF5. Eliminar usuario	RF74. Adicionar área	RF142. Insertar revista

Capítulo 2. Análisis del sistema

RF6. Exportar usuario a Excel	RF75. Modificar datos de área	RF143. Modificar datos de revista
RF7. Exportar usuario a pdf	RF76. Visualizar listado de áreas	RF144. Eliminar revista
RF8. Habilitar usuario	RF77. Eliminar área	RF145. Mostrar datos de revista
RF9. Asignar usuario al área CICE	RF78. Adicionar índice	RF146. Visualizar listado de revistas
RF10. Asignar rol de especialista del CICE	RF79. Modificar datos de índice	RF147. Exportar revista a pdf
RF11. Exportar listado de usuarios a excel	RF80. Visualizar listado de índices	RF148. Exportar revista a excel
RF12. Exportar listado de usuarios a pdf	RF81. Eliminar índices	RF149. Exportar listado de revistas a pdf
RF13. Realizar búsquedas	RF82. Adicionar frecuencia	RF150. Exportar listado de revistas a excel
RF14. Adicionar temática	RF83. Modificar datos de frecuencia	RF151. Insertar curso
RF15. Modificar datos de temática	RF84. Visualizar listado de frecuencias	RF152. Modificar datos de curso
RF16. Visualizar listado de temáticas	RF85. Eliminar frecuencias	RF153. Eliminar curso
RF17. Eliminar temática	RF86. Adicionar nivel	RF154. Mostrar datos de curso
RF18. Adicionar tipo de curso	RF87. Modificar datos de nivel	RF155. Visualizar listado de cursos
RF19. Modificar datos de tipo de curso	RF88. Visualizar listado de niveles	RF156. Exportar curso a pdf
RF20. Visualizar listado de tipos de cursos	RF89. Eliminar nivel	RF157. Exportar curso a excel

Capítulo 2. Análisis del sistema

RF21. Eliminar tipo de curso	RF90. Adicionar editorial	RF158. Exportar listado de cursos a pdf
RF22. Insertar noticia	RF91. Modificar datos de editorial	RF159. Exportar listado de cursos a excel
RF23. Modificar noticia	RF92. Visualizar listado de editoriales	RF160. Adicionar memoria de evento
RF24. Eliminar noticia	RF93. Eliminar editorial	RF161. Modificar memoria de evento
RF25. Mostrar noticia	RF94. Adicionar frecuencia	RF162. Eliminar memoria de evento
RF26. Visualizar listado de noticias	RF95. Modificar sección "datos generales" del sitio	RF163. Visualizar listado de memorias de eventos
RF27. Insertar curiosidad	RF96. Modificar sección "sobre nosotros" del sitio	RF164. Exportar memoria de evento a pdf
RF28. Modificar curiosidad	RF97. Modificar sección "objetivos" del sitio	RF165. Exportar memoria de evento a excel
RF29. Eliminar curiosidad	RF98. Modificar sección "misión" del sitio	RF166. Exportar listado de memorias de eventos a pdf
RF30. Mostrar curiosidad	RF99. Modificar sección "visión" del sitio	RF167. Exportar listado de memorias de eventos a excel
RF31. Visualizar listado de curiosidades	RF100. Modificar sección "estructura" del sitio	RF168. Adicionar publicación en revista
RF32. Insertar maestría	RF101. Adicionar galería	RF169. Modificar publicación en revista
RF33. Modificar datos de maestría	RF102. Modificar galería	RF170. Eliminar publicación en revista

Capítulo 2. Análisis del sistema

RF34. Eliminar maestría	RF103. Eliminar galería	RF171. Visualizar listado de publicaciones en revistas
RF35. Mostrar datos de maestría	RF104. Visualizar listado de galerías	RF172. Exportar publicación en revista a pdf
RF36. Visualizar listado de maestrías	RF105. Asignar visibilidad a galería	RF173. Exportar publicación en revista a excel
RF37. Insertar doctorado	RF106. Adicionar documentos al sitio	RF174. Exportar listado de publicaciones en revistas a pdf
RF38. Modificar datos de doctorado	RF107. Modificar documentos del sitio	RF175. Exportar listado de publicaciones en revistas a excel
RF39. Eliminar doctorado	RF108. Eliminar documentos del sitio	RF176. Adicionar participación en curso de posgrado impartido
RF40. Mostrar datos de doctorado	RF109. Visualizar listado de documentos del sitio	RF177. Modificar participación en curso de posgrado impartido
RF41. Visualizar listado de doctorados	RF110. Adicionar enlace a universidades	RF178. Visualizar listado de participación en cursos de posgrado impartidos
RF42. Insertar evento	RF111. Modificar enlace a universidades	RF179. Exportar participación en curso de posgrado impartido a pdf
RF43. Modificar datos de evento	RF112. Eliminar enlace a universidades	RF180. Exportar participación en curso de posgrado impartido a excel

Capítulo 2. Análisis del sistema

RF44. Eliminar evento	RF113. Visualizar listado de enlaces a universidades	RF181. Exportar listado de participación en cursos de posgrado impartidos pdf
RF45. Mostrar datos de evento	RF114. Insertar grupo	RF182. Exportar listado de participación en cursos de posgrado impartidos a excel
RF46. Visualizar listado de eventos	RF115. Modificar datos de grupo	RF183. Obtener reporte general de CTI a partir de indicadores en excel
RF47. Exportar evento a pdf	RF116. Eliminar grupo	RF184. Obtener reporte general de CTI a partir de indicadores en pdf
RF48. Exportar evento a excel	RF117. Mostrar datos de grupo	RF185. Adicionar memoria de evento en el perfil de usuario
RF49. Exportar listado de eventos a pdf	RF118. Visualizar listado de grupos	RF186. Modificar memoria de evento en el perfil de usuario
RF50. Exportar listado de eventos a excel	RF119. Asignar visibilidad de eventos en el portal	RF187. Eliminar memoria de evento en el perfil de usuario
RF51. Adicionar participación en curso de posgrado recibido	RF120. Adicionar participación en evento	RF188. Visualizar listado de memorias de eventos en el perfil de usuario
RF52. Modificar participación en curso de posgrado recibido	RF121. Modificar participación en evento	RF189. Adicionar publicación en revista en el perfil de usuario
RF53. Visualizar listado de participación en cursos de posgrado recibido	RF122. Eliminar participación en evento	RF190. Modificar publicación en revista en el perfil de usuario

Capítulo 2. Análisis del sistema

RF54. Exportar participación en curso de posgrado recibido a pdf	RF123. Visualizar listado de participación en eventos	RF191. Eliminar publicación en revista en el perfil de usuario
RF55. Exportar participación en curso de posgrado recibido a excel	RF124. Exportar participación en evento a pdf	RF192. Visualizar listado de publicaciones en revistas en el perfil de usuario
RF56. Exportar listado de participación en cursos de posgrado recibidos a pdf	RF125. Exportar participación en evento a excel	RF193. Adicionar participación en curso de posgrado impartido en el perfil de usuario
RF57. Exportar listado de participación en cursos de posgrado recibidos a excel	RF126. Exportar listado de participación en eventos a pdf	RF194. Modificar participación en curso de posgrado impartido en el perfil de usuario
RF58. Visualizar el perfil de usuario	RF127. Exportar listado de participación en eventos a excel	RF195. Visualizar listado de participación en cursos de posgrado impartidos en el perfil de usuario
RF59. Visualizar noticias destacadas en el portal	RF128. Adicionar participación en evento en el perfil de usuario	RF196. Adicionar participación en curso de posgrado recibido en el perfil de usuario
RF60. Visualizar avisos en el portal	RF129. Modificar participación en evento en el perfil de usuario	RF197. Modificar participación en curso de posgrado recibido en el perfil de usuario

Capítulo 2. Análisis del sistema

RF61. Visualizar en el portal los próximos eventos a desarrollarse	RF130. Eliminar participación en evento en el perfil de usuario	RF198. Visualizar listado de participación en cursos de posgrado recibido en el perfil de usuario
RF62. Visualizar curiosidades en el portal	RF131. Exportar perfil de usuario a pdf	RF199. Visualizar enlaces internacionales en el portal
RF63. Visualizar últimas noticias en el portal	RF132. Exportar perfil de usuario a excel	RF200. Visualizar enlaces a otras universidades en el portal
RF64. Visualizar comentarios en el portal	RF133. Modificar perfil de usuario	RF201. Visualizar estadísticas en el portal
RF65. Visualizar noticias destacadas en el portal	RF134. Enviar comentario en el portal	RF202. Visualizar video en el portal
RF66. Visualizar galería de imágenes en el portal	RF135. Visualizar datos de contacto del CICE	RF203. Visualizar estadísticas en la administración del portal
RF67. Visualizar datos del centro en el portal	RF136. Visualizar líneas de investigación del centro en el portal	RF204. Visualizar documentos del centro en el portal
RF68. Visualizar oferta académica	RF137. Visualizar enlaces nacionales en el portal	RF205. Visualizar listado de participación en eventos en el perfil de usuario
RF69. Visualizar cursos en el portal		

Capítulo 2. Análisis del sistema

2.1.2 Especificación de Requisitos no Funcionales

RNF1. Apariencia o interfaz externa: La interfaz externa del producto es de fácil navegación por el usuario, sencilla y legible. Su funcionamiento es intuitivo y requiere de información mínima. Se garantiza la uniformidad de las interfaces de usuario, teniendo en cuenta que las operaciones comparables siempre se activen de la misma forma. Los conceptos y términos que se plasman en las interfaces del sistema fueron obtenidos de la experiencia de las personas que más utilizarán este producto. Contiene mecanismos para recuperarse ante errores. De igual forma, al cometer errores, la interfaz proporciona características de ayuda. Se incluyen elementos visuales para la selección de información siempre que sea posible, para minimizar los posibles errores.

RNF2. Usabilidad: El sistema puede ser utilizado por cualquier usuario que posea conocimientos informáticos básicos. El software tiene una curva de aprendizaje baja, que permite al usuario familiarizarse rápidamente con los elementos del sistema y operarlo de forma correcta en poco tiempo de uso. Cumple con la regla de los tres clics para llegar a cada una de las funcionalidades del sistema y se incluyen “migas de pan” (*bread crumbs*) para indicar la ubicación exacta que tiene el recurso dentro del sistema.

RNF3. Requisitos de software:

Para las estaciones de trabajo:

Los ordenadores que serán utilizados por los usuarios del sistema para acceder a la aplicación y operar en la misma deberán tener los siguientes requerimientos de software:

- Un navegador Web Internet Explorer 8.0 o superior o Mozilla Firefox 15.0 o superior
- Sistemas Operativos Windows XP Service Pack 3 o GNU/Linux distribución Ubuntu 12.04

Para los servidores:

Servidor de Aplicación:

- Servidor Web Apache en su versión 2.0 o superior
- Sistema Operativo: GNU/ Linux Distribución Ubuntu 12.04 o Windows server 2003

Servidor de Base de Datos

- PostgreSQL en su versión 9.1 o superior, como Sistema Gestor de Bases de Datos
- Sistema Operativo: GNU/ Linux Distribución Ubuntu 12.04

RNF4. Requisitos de hardware:

Para las estaciones de trabajo:

- Tipo de procesador: Intel Pentium IV o superior
- Memoria RAM: 512 Mb o superior
- Velocidad del procesador: 1.5 GHz o superior
- Tarjeta de red

Para los servidores:

Servidor de Base de Datos

- Tipo de procesador: Intel Core2Duo o superior
- Memoria RAM: 2Gb o superior
- Velocidad del procesador: 3.00 GHz
- Disco duro: 160Gb o superior
- Tarjeta de red

Servidor de Aplicación

- Tipo de procesador: Intel Dual Core o superior
- Velocidad del procesador: 3.00 GHz
- Memoria RAM: 2 Gb o superior
- Disco duro: 80Gb o superior
- Tarjeta de red

RNF5. Rendimiento: El tiempo de respuesta ante diferentes peticiones de los usuarios será inferior a los tres (3) segundos.

RNF6. Requisitos de seguridad: El sistema establecerá una política de acceso basada en roles, usuarios y permisos de los usuarios para las operaciones.

Capítulo 2. Análisis del sistema

2.2 Modelación del Sistema

El Modelo de Casos de Uso del Sistema es un modelo que contiene actores, casos de uso y sus relaciones; describe lo que el sistema hace por sus usuarios y bajo qué restricciones. Permite que los desarrolladores y clientes lleguen a un acuerdo sobre los requerimientos, y proporciona la entrada fundamental para el diseño y las pruebas. (Jacobson, y otros, 2000)

2.2.1 Descripción de los actores que interactúan con el sistema

Un actor es un usuario del sistema. Esto incluye usuarios humanos y otros sistemas computacionales. El conjunto de casos de uso al que un actor tiene acceso define un rol en el sistema y el alcance de su acción. En la Tabla 2 se describen los actores que interactúan con las funcionalidades en la aplicación.

Tabla 2 Descripción de los actores del sistema

Actor del sistema	Descripción
Administrador	Es el encargado de gestionar los usuarios. Tiene acceso a todas las funcionalidades del sistema.
Especialista CICE	Es la persona encargada de gestionar toda la información referente a Ciencia y Técnica. Ejemplo de ello son los eventos, cursos, las revistas donde se publican las investigaciones que se realizan. Gestiona los nomencladores asociados a los eventos, revistas y cursos. También visualiza el expediente CTI_CICE. Tiene acceso a las funcionalidades del Usuario CICE.
Usuario CICE	Son las personas que trabajan en el centro. Tienen permisos para gestionar su participación en eventos, en cursos a ofertar y a recibir, las publicaciones que han realizado en revistas y en memorias de eventos. También visualizan el expediente CTI_Investigador.
Usuario	Es toda persona que interactúa con la aplicación. Podrá visualizar los cursos, eventos, noticias, comentar en el foro. También podrá exportar el listado de eventos y revistas.

Capítulo 2. Análisis del sistema

2.2.2 Estructuración del sistema

Los requisitos funcionales se agruparon en 62 casos de uso del sistema auxiliándose de los patrones de casos de uso (múltiples actores roles comunes, CRUD completo, extensión concreta), conformando así el modelo de casos de uso del sistema, en el cual intervienen 4 actores. Para un mejor entendimiento del sistema se agrupan estas funcionalidades en paquetes que guardan una relación estrecha, que se presenta a continuación:

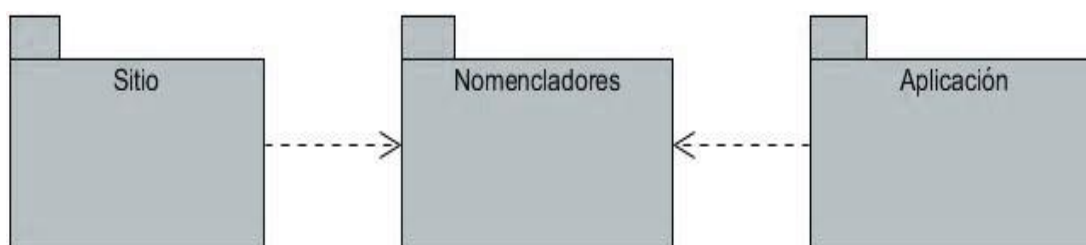


Figura 3 Estructura del sistema a través de paquetes

Diagrama de Casos de Uso del Sistema

Los diagramas de casos de uso documentan el comportamiento de un sistema desde el punto de vista del usuario. Por lo tanto, los casos de uso determinan los requisitos funcionales del sistema. Su ventaja principal es la facilidad para interpretarlos, lo que hace que sean especialmente útiles en la comunicación con el cliente (Cáceres, 2008). En la Figura 4 se representa el Diagrama de Casos de Uso del Sistema (DCUS).

Capítulo 2. Análisis del sistema

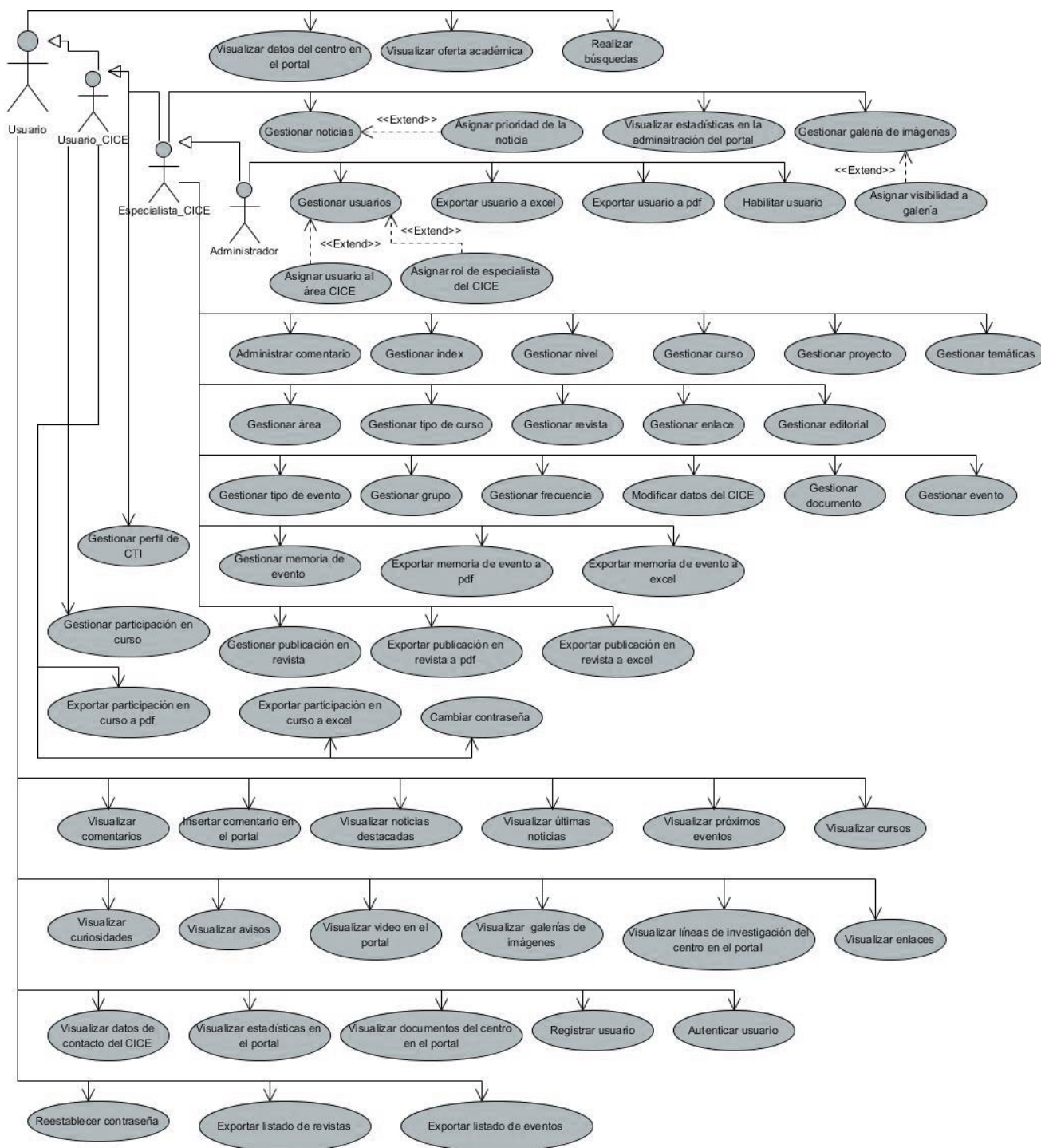


Figura 4 Diagrama de Casos de Uso del Sistema

Capítulo 2. Análisis del sistema

2.2.3 Descripción textual de los Casos de Uso del Sistema

Debido a que las descripciones textuales son muy extensas, en la Tabla 3 solo se muestra la correspondiente al caso de uso Gestionar Eventos. El resto de las descripciones textuales se encuentran en el Expediente de Proyecto.

Tabla 3 Descripción textual del Caso de Uso Gestionar Eventos

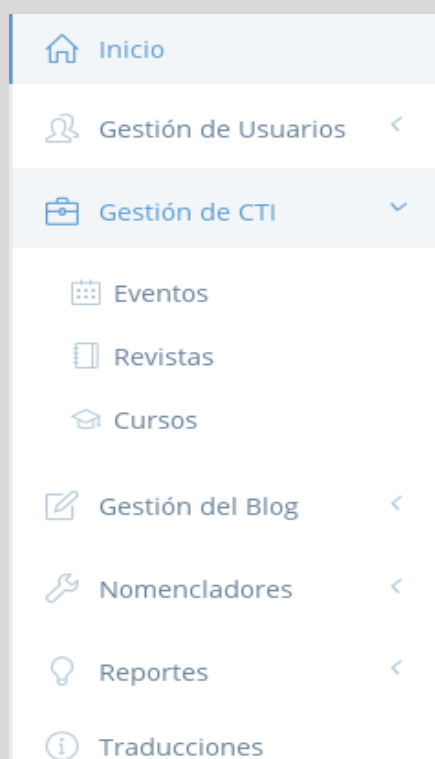
Caso de Uso	Gestionar Eventos
Actores	Especialista CICE
Propósito	Este caso de uso se realiza con el objetivo de gestionar los eventos en los que puede participar cualquier trabajador de la universidad.
Resumen	El caso de uso inicia cuando el Especialista CICE desea realizar algunas de las siguientes operaciones: Insertar, Modificar, Eliminar o Mostrar los datos asociados a los eventos que se ofrecen y termina cuando han sido completadas las acciones seleccionadas.
Complejidad	Alta
Referencias	RF42, RF43, RF44, RF45, RF46
Precondiciones	El actor tiene que estar autenticado.
Flujo Normal de Eventos	
Acciones del Actor	Respuesta del Sistema
1. El caso de uso se inicia cuando se selecciona la opción <i>Gestión de CTI</i> en la Página Principal. (Ver Interfaz 1.1)	2. Se despliegan las operaciones que forman parte de CTI que pueden ser gestionadas. (Ver Interfaz 1.1)
3. Selecciona la opción <i>Eventos</i> .	4. Muestra una interfaz Eventos (Ver Interfaz 1.2) que contiene: - El listado de los eventos registrados. - El lugar y la fecha de los eventos registrados.

Capítulo 2. Análisis del sistema

- Opciones de *Modificar*, *Eliminar* y *Mostrar* por cada uno de los eventos registrados.
- Botón *Crear Evento*.
- Un Textfield acompañado de un botón *Buscar*.
- Si el actor desea adicionar un evento, ir a la sección “Adicionar Evento”.
- Si el actor desea eliminar un evento ir a la sección “Eliminar Evento”.
- Si el actor desea modificar un evento, ir a la sección “Modificar Evento”.
- Si el actor desea mostrar los datos de un evento “Mostrar Datos de Evento”.

Prototipo de Interfaz

Interfaz 1.1



Capítulo 2. Análisis del sistema

Interfaz 1.2

Eventos				
<input type="text" value="Buscar"/> <input type="button" value="Q Buscar"/> <input type="button" value="+ Crear Evento"/>				
Nombre	Lugar	Visible en Portada	Fecha del Evento	Acciones
ECML-PKDD 2016 : European Conference on Machine Learning and Pri...	Riva del Garda	<input checked="" type="checkbox"/>	19/09/2016	
ECML-PKDD 2016 : European Conference on Machine Learning and Pri...	Riva del Garda	<input checked="" type="checkbox"/>	19/09/2016	
ICONIP 2016 : The 23rd International Conference on Neural Inform...	Kyoto	<input checked="" type="checkbox"/>	16/10/2016	
Congreso Internacional de Educación y Aprendizaje	Vancouver	<input checked="" type="checkbox"/>	13/07/2016	
ICML 2016 : 33rd International Conference on Machine Learning (I...	New York	<input checked="" type="checkbox"/>	19/06/2016	

Sección "Adicionar Evento"

Acciones del Actor	Respuesta del Sistema
1. Da clic en la opción <i>Crear Evento</i> (Ver interfaz 1.3)	2. Muestra una ventana para Adicionar un evento: (Ver interfaz 1.4). Esta contiene los siguientes campos: Imagen, Nombre del evento, Tipo (Provincial, Nacional, Internacional), País, Lugar, Url, Temáticas, Visible en Portada (Sí o No), fecha del evento, fecha de entrega de resúmenes, fecha de entrega de trabajo, programa del evento, plantilla, descripción. Un botón: <i>Cancelar</i> y <i>Guardar</i> .
3. Introduce los datos que se solicitan, siendo todos los campos obligatorios excepto la URL y da clic en el botón <i>Guardar</i> .	4. El caso de uso termina cuando adiciona un evento y muestra la interfaz <i>Eventos</i> actualizada.

Flujo Alternativo

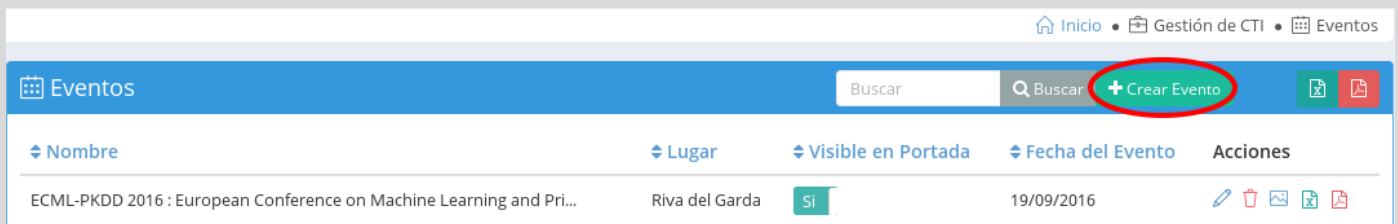
3.1 No introduce alguno de los datos obligatorios y da clic en el botón <i>Guardar</i> .	3.2 Indica que el actor debe llenar los campos obligatorios para que se pueda ejecutar la acción. (Ver interfaz 1.5). Se realiza el paso 3 de la Sección "Adicionar Evento".
--	--

Capítulo 2. Análisis del sistema

3.1 Introduce alguno de los datos que se solicitan con error y da clic en el botón <i>Guardar</i> .	3.2 Indica al actor el campo que tiene error. (Ver interfaz 1.6). Se realiza el paso 3 de la Sección “Adicionar Evento”.
3.1 Introduce los datos que se solicitan y da clic en el botón <i>Cancelar</i> .	3.2 Cierra la interfaz de <i>Adicionar Evento</i> sin realizar cambios y muestra la interfaz <i>Eventos</i> .

Prototipo de Interfaz

Interfaz 1.3



Interfaz 1.4

Capítulo 2. Análisis del sistema

Crear Evento

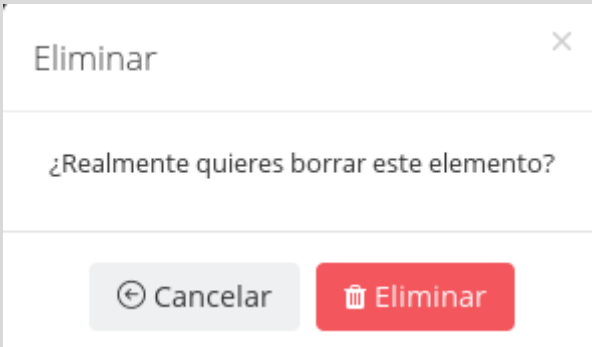
Imagen <input type="text"/>	Nombre <input type="text"/>
Fecha del Evento <input type="text"/>	Tipo Seleccione...
Fecha de Entrega de Resúmenes <input type="text"/>	País Seleccione...
Fecha de Entrega de Trabajo <input type="text"/>	Lugar <input type="text"/>
Programa del Evento <input type="text"/> Examinar...	Url <input type="text"/>
Plantilla <input type="text"/> Examinar...	Temáticas Seleccione...
Nivel Seleccione...	Visible en Portada Si
Descripción <div><p>B I S Source</p><hr/><div style="border: 1px solid #ccc; height: 150px;"></div></div>	Correo Electrónico <input type="text"/>

Interfaz 1.5

Capítulo 2. Análisis del sistema


Interfaz 1.6	
<div style="background-color: #007bff; color: white; padding: 5px;"> 📅 Crear Evento </div> <div style="display: flex; justify-content: space-between; margin-top: 10px;"> <div style="width: 45%;"> <p>Imagen</p> <div style="border: 1px solid #ccc; width: 100px; height: 60px; margin-bottom: 5px;"></div> <div style="background-color: #007bff; color: white; width: 20px; height: 20px; display: flex; align-items: center; justify-content: center; margin-left: auto;">↻</div> <p style="font-size: 0.8em; color: #dc3545;">Por favor introduce un valor</p> </div> <div style="width: 45%;"> <p>Nombre</p> <div style="border: 1px solid #dc3545; width: 100%; height: 25px; margin-bottom: 5px;"></div> <p style="font-size: 0.8em; color: #dc3545;">Por favor introduce un valor</p> <p>Tipo</p> <div style="border: 1px solid #28a745; padding: 2px; margin-bottom: 5px;">Nacional ✔</div> <p>País</p> <div style="border: 1px solid #28a745; padding: 2px; margin-bottom: 5px;">Cuba ✔</div> <p>Lugar</p> <div style="border: 1px solid #28a745; padding: 2px; margin-bottom: 5px;">Habana ✔</div> </div> </div> <div style="margin-top: 10px;"> <p>Fecha del Evento</p> <div style="border: 1px solid #28a745; padding: 2px; display: flex; justify-content: space-between; align-items: center;"> 17-06-2016 ✔ 📅 </div> <p>Fecha de Entrega de Resúmenes</p> <div style="border: 1px solid #28a745; padding: 2px; display: flex; justify-content: space-between; align-items: center;"> 05-04-2016 ✔ 📅 </div> </div>	
Sección “Eliminar Evento”	
Acciones del Actor	Respuesta del Sistema
1. Selecciona el evento que desea eliminar del listado que se muestra en la interfaz <i>Eventos</i> y da clic en el botón <i>Eliminar</i> .	2. Muestra una ventana al actor donde le indica que debe confirmar si está seguro que desea eliminar el evento seleccionado. (Ver interfaz 1.7)
3. Da clic en el botón <i>Eliminar</i> .	4. El caso de uso termina cuando elimina el evento seleccionado por el actor y muestra la interfaz <i>Eventos</i> actualizada.
Flujos Alternos	
3.1 Da clic en el botón <i>Cancelar</i> .	3.2 Cierra la ventana sin realizar cambios y muestra la interfaz <i>Eventos</i> .
Prototipo de Interfaz	

Capítulo 2. Análisis del sistema

Interfaz 1.7	
	
Sección “Modificar Evento”	
Acciones del Actor	Respuesta del Sistema
1. Selecciona el evento al cual desea modificar sus datos a partir del listado que se muestra en la interfaz <i>Eventos</i> y da clic en el botón <i>Modificar</i> .	2. Muestra una ventana con los datos del evento seleccionado, (Ver interfaz 1.8). Contiene: Tres botones: <i>Guardar</i> , <i>Eliminar</i> y <i>Cancelar</i> .
3. Introduce los nuevos datos y da clic en el botón <i>Guardar</i> .	4. El caso de uso termina cuando actualiza los datos del evento y muestra la interfaz <i>Eventos</i> actualizada.
Flujos Alternos	
3.1 No introduce alguno de los datos obligatorios asociados al evento que se encuentra modificando y da clic en el botón <i>Guardar</i> .	3.2 Indica al actor que debe llenar los campos faltantes para que se pueda ejecutar la acción. Se realiza el paso 3 de la Sección “Modificar Evento”.
3.1 Introduce alguno de los datos nuevos con error y da clic en el botón <i>Guardar</i> .	3.2 Indica al actor el campo que tiene error. Se realiza el paso 3 de la Sección “Modificar Evento”.
3.1 Introduce los nuevos datos y da clic en el botón <i>Cancelar</i> .	3.2 No realiza cambios, se cierra la ventana y se muestra la interfaz <i>Eventos</i> .
Prototipo de Interfaz	
Interfaz 1.8	

Capítulo 2. Análisis del sistema

Modificar Evento (ICONIP 2016 : The 23rd International Conference on Neural Information Processing)...

Imagen 	Nombre ICONIP 2016 : The 23rd International Conference on Neural Inform
Fecha del Evento 16-10-2016	Tipo Internacional en el extranjero
Fecha de Entrega de Resúmenes 24-04-2016	País Japón
Fecha de Entrega de Trabajo 15-05-2016	Lugar Kyoto
Programa del Evento Examinar...	Url http://iconip2016.org
Plantilla Examinar...	Temáticas x Ciencias de la información. x Redes y Sistemas de Información. x Bibliotecas y Archivos; Tratamiento, análisis y gestión de la Información.
Nivel Seleccione...	Visible en Portada Si
Descripción <p>La 23ª Conferencia Internacional sobre Neural Information Processing (ICONIP 2016) se llevará a cabo en Kyoto, una hermosa ciudad antigua capital de Japón, del 16 al 21 de octubre de 2016. Allí, el comité organizador tiene previsto varios nuevos eventos: exposiciones, sesiones especiales, joven investigador / estudiantes premios, talleres, etc. por favor, participen en la nueva ICONIP.</p>	Correo Electrónico

Guardar **Eliminar** **Cancelar**

Capítulo 2. Análisis del sistema

Sección “Mostrar Datos de Evento”


Acciones del Actor	Respuesta del Sistema
<p>1. Selecciona el evento del cual desea visualizar sus datos a partir del listado que se muestra en la interfaz <i>Eventos</i> y da clic en el botón <i>Mostrar</i>.</p>	<p>2. Muestra una ventana que contiene los datos del evento seleccionado. Ver interfaz 1.9. Dos botones: <i>Modificar</i>, <i>Eliminar</i> y <i>Cancelar</i>.</p>
<p>3. Si selecciona la opción <i>Modificar</i>, ver el paso 3 de la sección “Modificar Evento”.</p> <ul style="list-style-type: none"> - Si selecciona la opción <i>Eliminar</i>, ver el paso 2 de la sección “Eliminar Evento”. - Si desea regresar a la pantalla principal <i>Eventos</i>, da clic en la opción <i>Cancelar</i>. 	

Prototipo de Interfaz

Interfaz 1.9

Mostrar Evento (ECML-PKDD 2016 : European Conference on Machine Learning and Principles and Practice of Knowle...

Imagen



Fecha del Evento

Fecha de Entrega de Resúmenes

Fecha de Entrega de Trabajo

Programa del Evento

Plantilla

Temáticas

- Recursos electrónicos
- Ciencias Tecnológicas

Nivel

Descripción

La Conferencia Europea sobre aprendizaje automático y Principios y Práctica de Descubrimiento de Conocimiento se llevará a cabo en Riva del Garda, Italia, durante el 19-23 de septiembre de 2016. Este evento es la conferencia más importante de aprendizaje automático y minería de datos europea y se basa en una serie muy exitosa de 26 y 19 CELM PKDD conferencias, que se han organizado en forma conjunta por los últimos 15 años.

Nombre

Tipo

País

Lugar

Url

Visible en Portada

Correo Electrónico

[✎ Modificar](#)
[🗑 Eliminar](#)
[🚫 Cancelar](#)
[📄 Exportar a Excel](#)

Conclusiones parciales

A partir del desarrollo del presente capítulo se arribaron a las siguientes conclusiones:

- Con la identificación de los principales procesos, para gestionar la información asociada a los servicios que se brindan en el Centro de Innovación y Calidad de la Educación, como la gestión de la formación del pregrado y posgrado, de CTI, de los eventos, revistas y de identificación general del CICE, se logra determinar el objeto de automatización compuesto por los mismos procesos.
- Se determinaron las diferentes funcionalidades y cualidades que el sistema debe cumplir mediante la especificación de requisitos. Se identificaron 205 requisitos funcionales que fueron agrupados en 62 casos de uso del sistema.
- Se realizó la descripción textual de cada caso de uso, lo cual permitió tener un mejor entendimiento de las funcionalidades que el sistema debe tener, el comportamiento del mismo y las secuencias de actividades que debe realizar el usuario para lograr su objetivo.

Capítulo 3. Descripción de la solución propuesta

Cuando se inicia la construcción de un sistema de software, luego de conocer cuáles son las necesidades básicas del usuario final, el ambiente donde será desarrollado el mismo, las condiciones para ello, las tecnologías y recursos necesarios para lograr un ambiente de trabajo adecuado, entonces se establecen los elementos fundamentales de la línea base para la arquitectura tecnológica del sistema.

En el presente capítulo se realiza una descripción de la arquitectura definida y los patrones de diseño que se utilizan. Se muestran los diagramas propuestos por la metodología, apoyándose en modelos que permiten la comprensión del diseño propuesto, así como el modelo de datos empleado.

3.1 Arquitectura de Software

La arquitectura del software alude a “*la estructura general de éste y a las formas en las que ésta da integridad conceptual a un sistema*”. En su forma más sencilla, la arquitectura es la estructura de organización de los componentes de un programa (módulos), la forma en la que éstos interactúan y la estructura de datos que utilizan. Sin embargo, en un sentido más amplio, los componentes se generalizan para que representen los elementos de un sistema grande y sus interacciones (Pressman, 2008)

Según el estándar 1471-2000 de la IEEE (Institute of Electrical and Electronics Engineers): «*La Arquitectura de Software es la organización fundamental de un sistema encarnada en sus componentes, las relaciones entre ellos y el ambiente y los principios que orientan su diseño y evolución*».

Es decir, la arquitectura brinda una visión global del sistema. Esto permite entenderlo, organizar su desarrollo, plantear la reutilización del software y hacerlo evolucionar. Se puede ver que la noción clave de la arquitectura es la organización y está relacionada con aspectos de rendimiento, usabilidad, reutilización, limitaciones económicas y tecnológicas. Ésta se ocupa de componentes y no de procedimientos; de las interacciones entre esos componentes y no de las interfaces; de las restricciones a ejercer sobre los componentes y las interacciones y no de los algoritmos, los procedimientos y los tipos. (Almerira, y otros, 2007)

3.1.1 Estilo de Arquitectura

Un estilo arquitectónico o variante arquitectónica define a una familia de sistemas informáticos en términos de su organización estructural. Un estilo arquitectónico describe componentes y las relaciones entre ellos con las restricciones de su aplicación, la composición asociada y el diseño para su construcción.

Para el desarrollo del sistema se seleccionó el estilo Llamada y Retorno. El sistema se constituye de un programa principal que tiene el control y varios subprogramas que se comunican con éste mediante el uso de llamadas.

El modelo utilizado es el de *Cliente – Servidor*. Es un modelo para el desarrollo de sistemas de información en el que las transacciones se dividen en procesos independientes que cooperan entre sí para intercambiar información, servicios o recursos (Fowler, y otros, 2002). Se denomina cliente al proceso que inicia el diálogo o solicita los recursos y servidor al proceso que responde a las solicitudes.

3.1.2 Patrón arquitectónico Modelo-Vista-Controlador

Los patrones arquitectónicos definen la estructura general del software, indican las relaciones entre los subsistemas y los componentes del software y definen las reglas para especificar las relaciones entre los elementos (clases, paquetes, subsistemas) de la arquitectura (Pressman, 2008).

El principio más importante de la arquitectura MVC es la separación del código en tres capas, dependiendo de su naturaleza. La lógica relacionada con los datos se incluye en el modelo, el código de la presentación en la vista y la lógica de la aplicación en el controlador. La programación se puede simplificar si se utilizan otros patrones de diseño. De esta forma, las capas del modelo, la vista y el controlador se pueden subdividir en más capas (Potencier, y otros, 2008). La Figura 5 muestra el funcionamiento interno de Symfony2 con la utilización del patrón arquitectónico MVC.

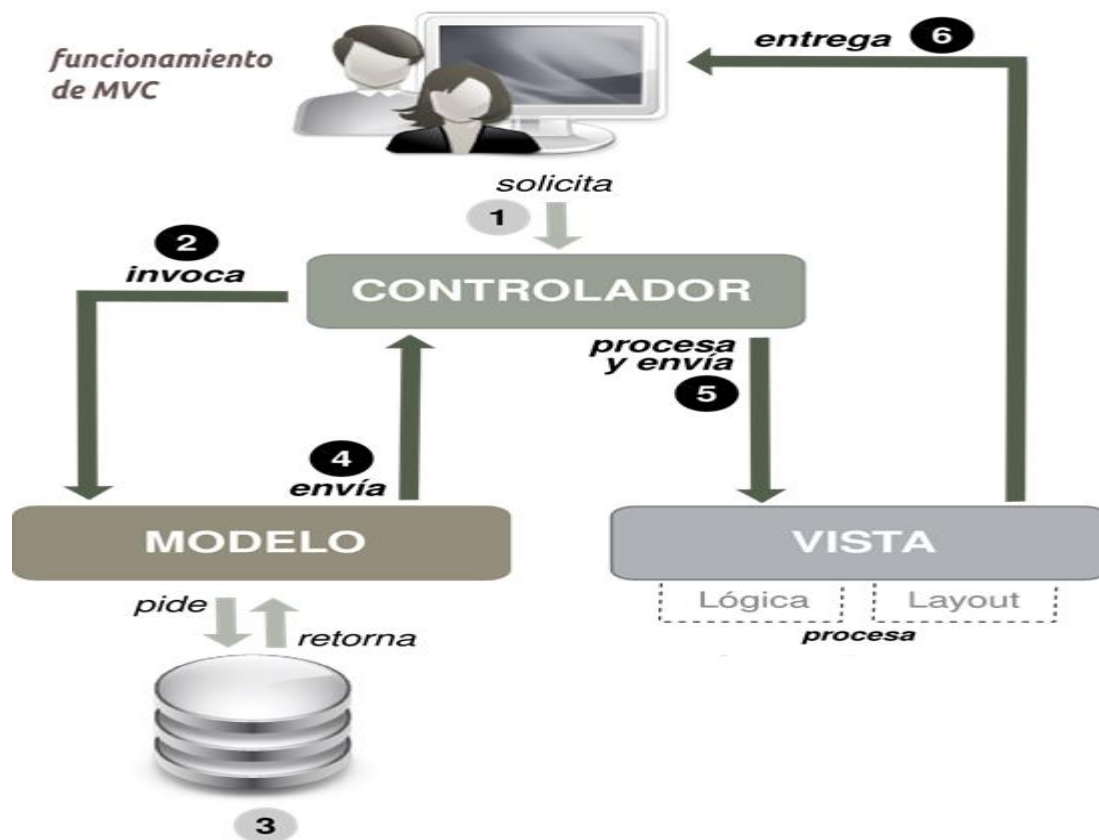


Figura 5 Funcionamiento interno de Symfony con el patrón MVC

Modelo: Es la representación de la información que maneja la aplicación. El modelo en sí son los datos puros que puestos en contexto del sistema proveen de información al usuario o a la aplicación misma. En Symfony, el acceso y la modificación de los datos se realizan mediante objetos. Doctrine2 es el motor que se encarga de esta generación automática para construir sus clases, crea la estructura y genera el código de las mismas. Las clases y archivos relacionados con el modelo se guardan en el directorio `src/NombreBundle/Entity/`.

Vista: Es la representación del modelo en forma gráfica disponible para la interacción con el usuario. En el caso de una aplicación Web, la "Vista" es una página HTML con contenido dinámico sobre el cuál el usuario puede realizar operaciones. La vista en Symfony2 está formada por plantillas twig (Es un motor de plantillas para el lenguaje de programación PHP) que se almacenan en el directorio `src/NombreBundle/Resources/Views/`.

Capítulo 3. Diseño del Sistema

Controlador: Es la capa encargada de manejar y responder las solicitudes del usuario, procesa la información necesaria y modifica el Modelo en caso de ser necesario. En Symfony2 todas las peticiones se realizan a través del controlador frontal `app.php`, el cual a través del enrutamiento delega las responsabilidades en las acciones que se implementan en las clases `Controller` de cada bundle.

3.1.3 Vista lógica

Describe la estructura interna del sistema, es decir, las partes que lo integran y cómo interactúan. Ilustra las realizaciones de guiones de uso clave, subsistemas, paquetes y clases que abarcan el comportamiento significativo arquitectónicamente.

En la Figura 6 se representa la vista lógica extendida del sistema propuesto, teniendo en cuenta los principales elementos organizativos de Symfony y en la Figura 7 se representa una vista lógica contextualizada de la aplicación a construir.

Modelo, Vista y Controlador: representan paquetes lógicos en correspondencia con la arquitectura definida.

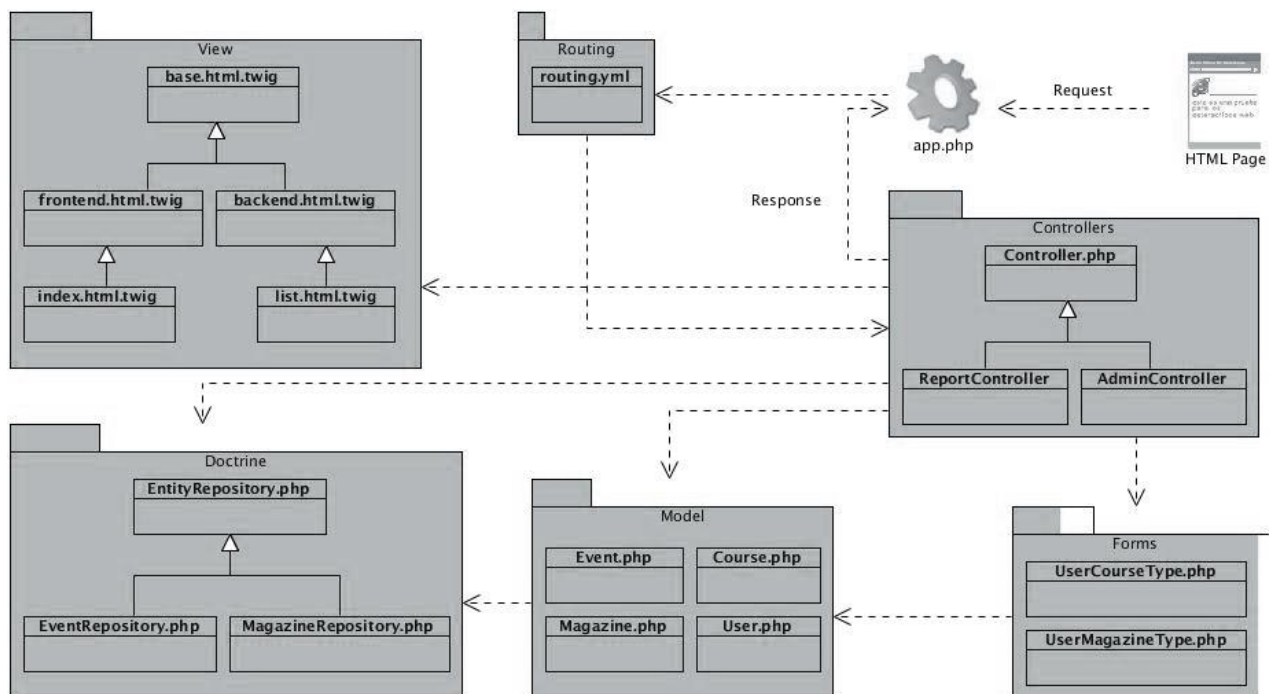


Figura 6 Vista lógica extendida

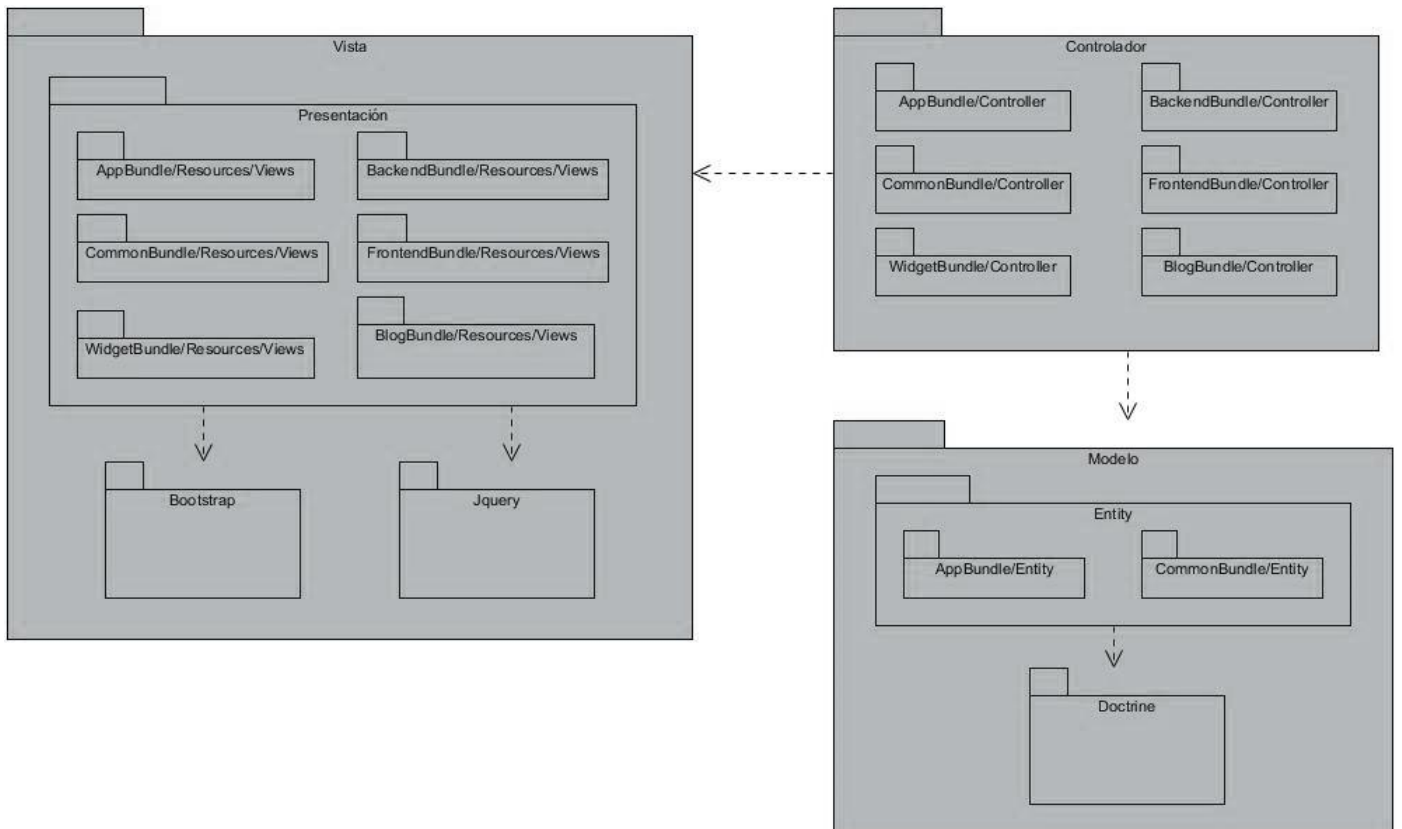


Figura 7 Vista Lógica de la aplicación

3.1.4 Diagrama de clases del diseño

Los diagramas de clases del diseño (DCD), corresponden a los casos de uso del sistema y muestran el diseño estático a través de las clases, subsistemas y relaciones que participan. Las clases del diseño representan una abstracción de una o varias clases en la implementación del sistema. El lenguaje utilizado para especificar una clase del diseño es el mismo que el lenguaje de programación utilizado, los métodos tienen correspondencia directa con el debido método de la implementación de clases.

Capítulo 3. Diseño del Sistema

En la Figura 8 se muestra el Diagrama de Clases del Diseño del Caso de Uso Gestionar Evento.

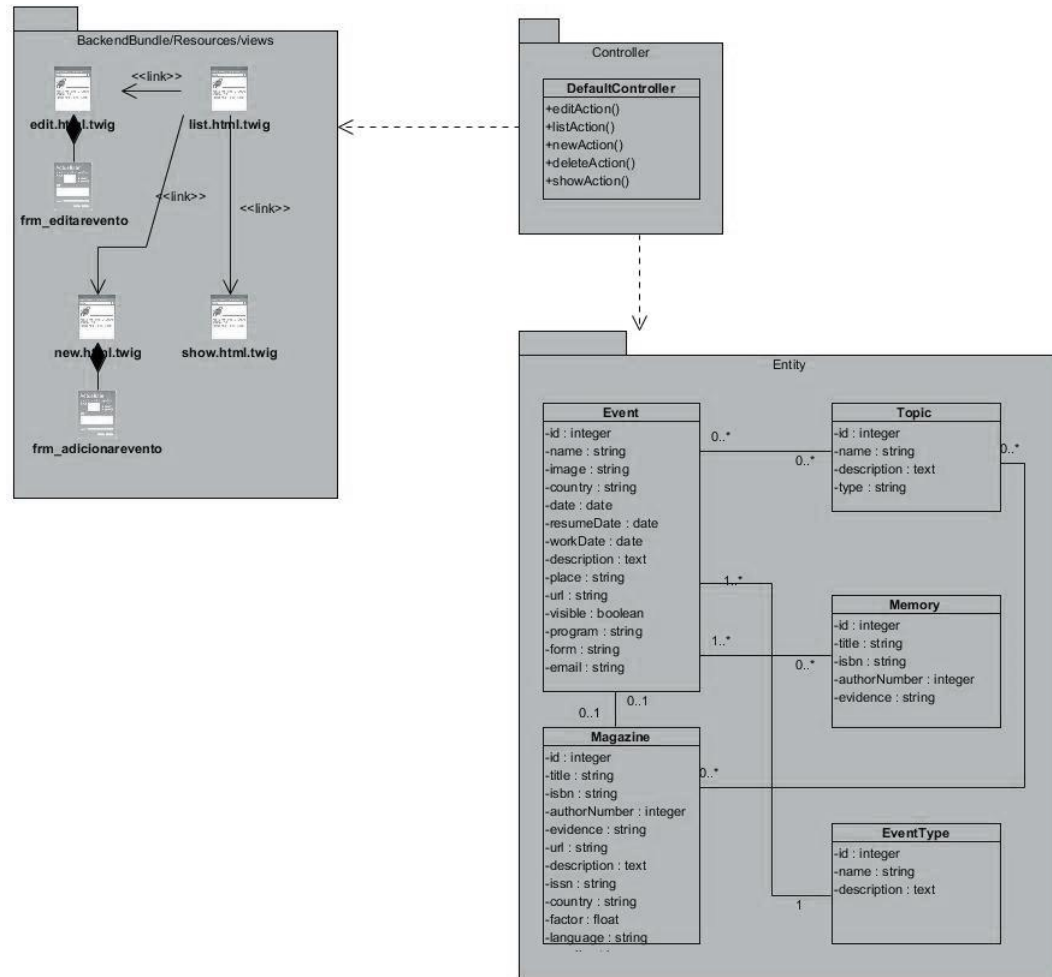


Figura 8 Diagrama de Clases del Diseño del Caso de Uso Gestionar Evento

3.1.5 Patrones del Diseño

Los patrones de diseño representan soluciones a problemas que surgen cuando se desarrolla un software en un contexto particular. Un patrón de diseño es una descripción de clases y objetos comunicándose entre sí adaptada para resolver un problema de diseño general en un contexto particular (Larman, 1999).

- **Patrones GRASP**

Los Patrones Generales de Software para Asignar Responsabilidades (GRASP por sus siglas en inglés), son parejas de problema y solución con un nombre, que codifican buenos principios y sugerencias

Capítulo 3. Diseño del Sistema

relacionados frecuentemente con la asignación de responsabilidades. Describen los principios fundamentales de la asignación de responsabilidades a objetos, expresados en forma de patrones. Los patrones GRASP utilizados en la investigación son:

Alta Cohesión: la cohesión es una medida de cuanto están relacionadas y enfocadas las responsabilidades de una clase. El uso del *framework* Symfony provee mediante una alta cohesión la asignación de responsabilidades y el empaquetamiento de archivos. Se evidencia en las clases controladoras debido a que Symfony implementa una clase controladora por cada entidad, lo que garantiza que se trabaje sobre una sola área de aplicación.

Experto: permite asignar una responsabilidad al experto en información, a la clase que cuenta con la información necesaria para cumplir la responsabilidad. Symfony2 incluye la librería ORM Doctrine como interfaz de comunicación con las clases del modelo, lo que permite encapsular toda la lógica de los datos y generar clases para manipular la información de las entidades de la base de datos. Este patrón se evidencia en las entidades que son las que conforman el modelo y se encuentran en la carpeta *Entity* de cada *Bundle*. Son expertas en la información que manejan, por ejemplo: la entidad *Event* tiene toda la información asociada a un evento.

Bajo Acoplamiento: el propósito de este patrón es que exista entre las clases la menor dependencia posible, para en caso de producirse una modificación en alguna, no afecte las restantes. Este patrón se evidencia durante la creación de objetos de las entidades desde los controladores, que heredan únicamente de la clase *Controller* que provee el *framework*. El propio patrón arquitectónico abstrae la vista y el controlador del modelo, aportando una baja dependencia entre las clases.

Creador: tiene en cuenta para la asignación de responsabilidades a las clases relacionadas con la creación de objetos, de forma tal que una instancia de un objeto solo pueda ser creada por el objeto que contiene la información necesaria. El uso de este patrón permite crear las dependencias mínimas necesarias entre las clases, lo cual favorece al mantenimiento del sistema. Se evidencia en las clases *Controller* porque contienen la información necesaria para crear instancias de una entidad.

Controlador: todas las peticiones desde la web son manipuladas por un solo controlador: `app_dev.php` para el entorno de desarrollo o `app.php` para el entorno de producción, que es el punto de entrada único para la aplicación dentro del entorno correspondiente. Este patrón plantea que se debe asignar la

Capítulo 3. Diseño del Sistema

responsabilidad a una clase del manejo de los eventos de un sistema. Sirve de intermediario entre una clase interfaz y la clase que contiene el algoritmo de la funcionalidad. El controlador es el encargado de procesar las diferentes peticiones hechas por el cliente. Este patrón se evidencia en las clases controladoras, pues son las encargadas de manejar las peticiones de los usuarios, obtener la información que necesitan, para lo cual utilizan *Doctrine* para conectarse con la Base de Datos y enviar a la vista los datos para que sea observable por el usuario.

- **Patrones GoF**

Los patrones GoF (Gang of Four, en español: Pandilla de los Cuatro) se clasifican en tres (3) categorías basadas en su propósito: creacionales, estructurales y de comportamiento. En esta investigación se utiliza el patrón decorador.

Decorador: añade responsabilidades a un objeto de una forma dinámica y transparente, brinda una alternativa flexible a la herencia. En Symfony2 las vistas se encuentran en la carpeta *Resources/Views* de cada *bundle*. Se recomienda usar herencia en tres niveles donde en la base se tiene todo el código HTML que es dinámico para cada una de las páginas de la aplicación. En el segundo nivel se tienen los *layout frontend* y *backend* que contienen los elementos visuales que se repiten, como el menú lateral, el *footer*. En el tercer nivel se encuentran los elementos visuales que cambian según la información que se desea mostrar.

3.2 Diseño de la Base de Datos

Las bases de datos o banco de datos es un conjunto de datos pertenecientes a un mismo contexto y almacenados sistemáticamente para su posterior uso. Es una colección de información organizada de forma que un programa de ordenador pueda seleccionar rápidamente los fragmentos de datos que necesite. Una base de datos correctamente diseñada permite obtener acceso a la información exacta y actualizada (Hernandez, 2010).

El modelo de datos constituye la representación de un fenómeno de la realidad objetiva a través de los objetos, sus propiedades y las relaciones que se establecen entre ellos. El mismo está compuesto por tres piezas fundamentales: el objeto de datos, los atributos y las relaciones entre las que se conectan. En la Figura 9 se representa: el Diagrama Entidad-Relación. De igual forma en la Figura 16 se muestra el Diagrama de Clases Persistentes (Ver Anexo 2).

Capítulo 3. Diseño del Sistema

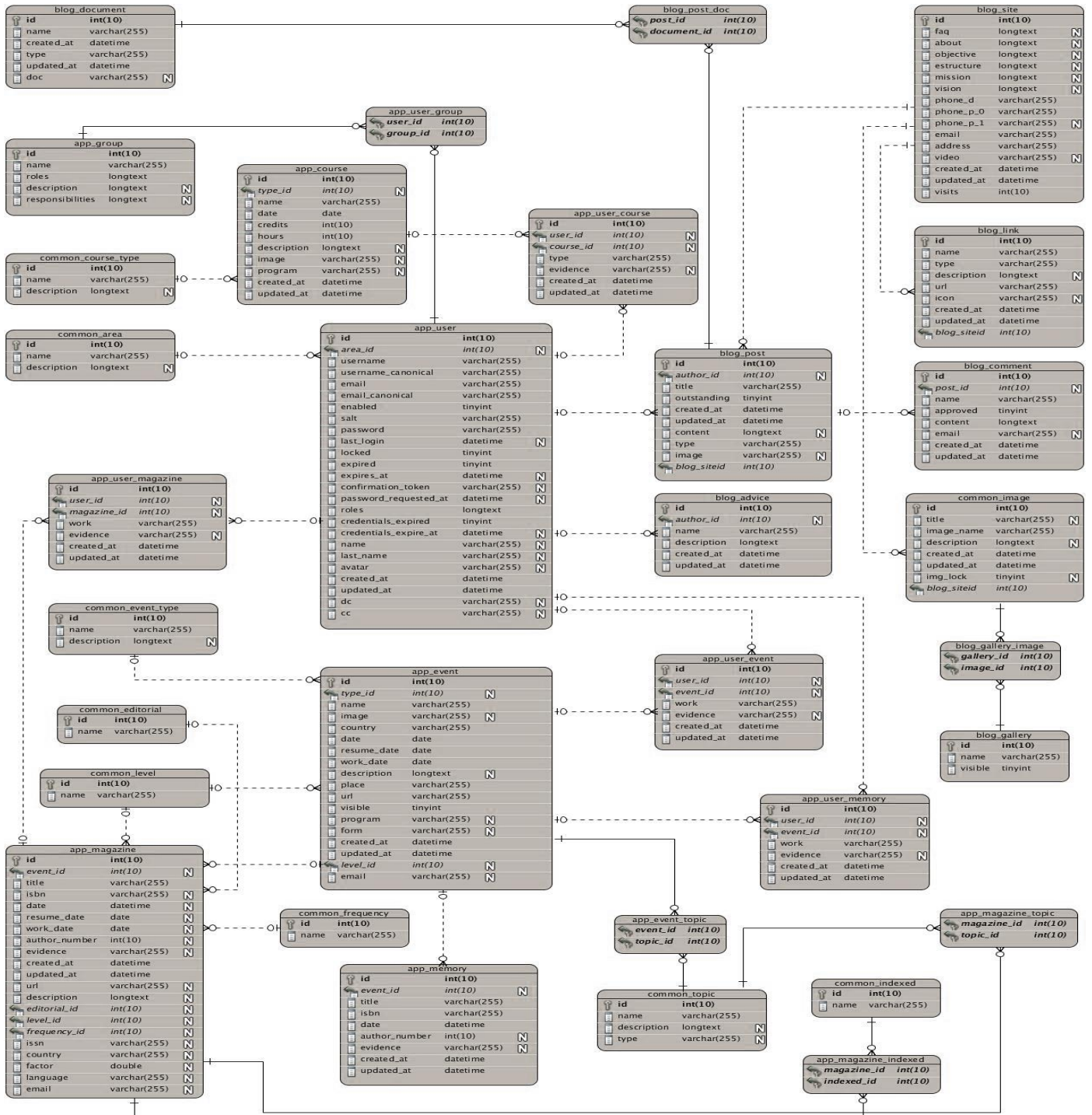


Figura 9 Diagrama entidad-relación

3.3 Modelo de despliegue

El modelo de despliegue es un modelo de objetos que describe la distribución física del sistema en términos de cómo se distribuye la funcionalidad entre los nodos de cómputo. (Jacobson, y otros, 2000). En la Figura 10 se muestra el Diagrama de Despliegue del sistema.

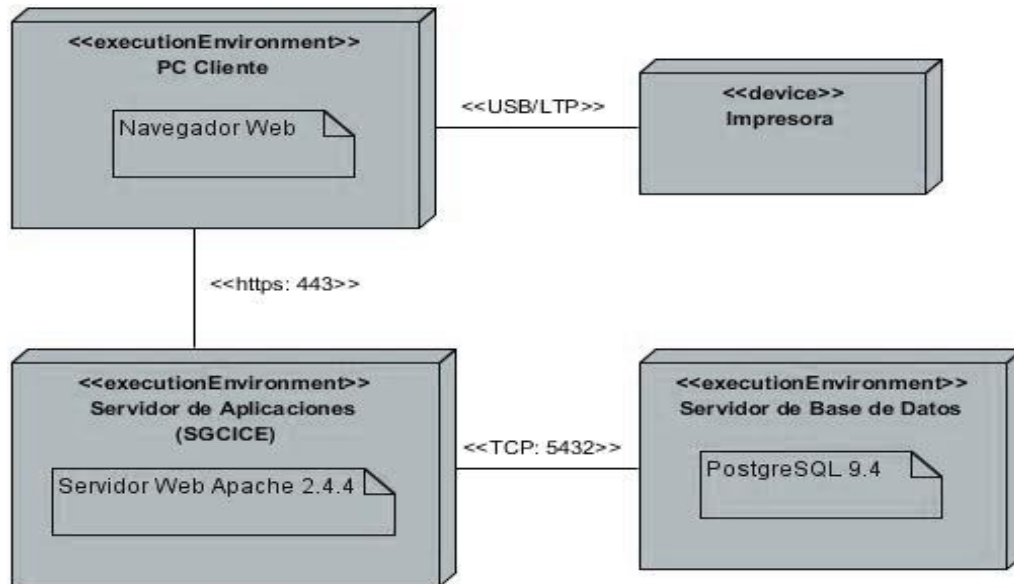


Figura 10 Diagrama de Despliegue

Descripción de los Nodos.

- `<<executionEnvironment>>` Servidor de Base de Datos: En este nodo se almacenan los datos del sistema.
- `<<executionEnvironment>>` Servidor de Aplicación Web SGCICE: Este nodo es el encargado de tener instalado el sistema al que tendrán acceso los usuarios desde las estaciones de trabajo.
- `<<executionEnvironment>>` PC Cliente: Este nodo representa la estación de trabajo que permite al usuario mediante el protocolo HTTPS y el puerto 443 acceder a la aplicación.
- `<<device>>` Impresora: Este nodo representa la impresora a través de la cual los usuarios pueden imprimir los resultados de las búsquedas y reportes generados.

Conclusiones parciales

A partir del desarrollo del presente capítulo se arribaron a las siguientes conclusiones:

- La definición del patrón arquitectónico Modelo – Vista – Controlador y los patrones de diseño GRASP y GoF empleados, aseguró una adecuada estructuración del sistema, haciendo más fácil la integración de los componentes del mismo, su actualización, corrección y mejora. De igual forma garantizó una mayor simplicidad en el desarrollo, así como la escalabilidad.
- Con la realización de la vista lógica extendida y contextualizada de la aplicación, así como de los diagramas de clases del diseño de cada uno de los casos de uso del sistema, se obtuvo una visión más exacta del sistema en términos de implementación, siendo de gran ayuda para el desarrollador.
- El modelo físico constituye la base persistente final para el almacenamiento eficiente de la información.
- La realización del modelo de despliegue brinda una distribución completa del acople de los distintos componentes por lo que está compuesto el sistema.

Capítulo 4. Implementación y Pruebas

La prueba es un elemento crítico para la garantía de la calidad del software, es el proceso que permite verificar y revelar la calidad de un producto de software. Son utilizadas para identificar posibles fallos de implementación, calidad o usabilidad de un programa. Básicamente es una de las fases del desarrollo de software consistente en probar las aplicaciones construidas. Las pruebas de software se integran dentro de las diferentes fases del ciclo del software dentro de la Ingeniería de Software (Jacobson, y otros, 2000). En el capítulo se plasma una descripción de los flujos de trabajo de implementación y pruebas realizadas al sistema. Se muestra un diagrama de componentes donde se describen los elementos físicos del sistema y sus relaciones. Finalmente se especifican las pruebas realizadas y los resultados alcanzados.

4.1 Diagrama de componentes

El diagrama de componentes es uno de los principales artefactos generados durante la implementación. Entre sus principales objetivos comprende: mostrar la dependencia lógica entre los distintos componentes del software y representar las relaciones entre los elementos que forman el código del sistema implementado. Estos esquemas describen unidades físicas del sistema y las relaciones existentes dentro del mismo. Cada unidad puede representar archivos simples, paquetes y librerías. (Microsoft, 2014). En la Figura 11 se muestra el Diagrama de Componentes del Caso de Uso Gestionar Evento.

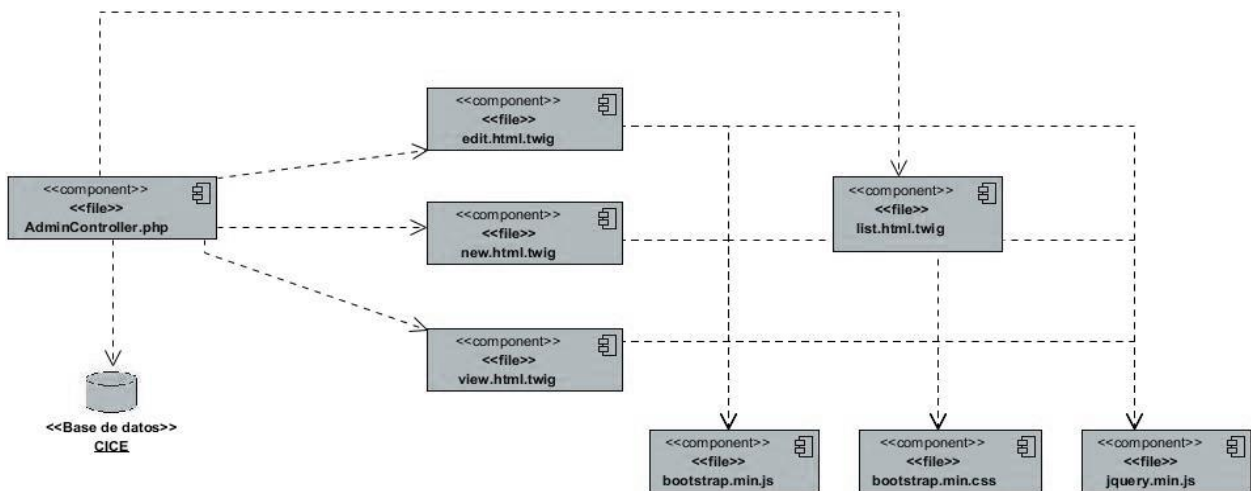


Figura 11 Diagrama de Componentes del Caso de Uso Gestionar Evento

4.2 Estándares de codificación

Un estándar de codificación comprende todos los aspectos de la generación de código. Un código fuente completo debe quedar como si un único programador hubiera escrito todo el código de una sola vez. Al inicio de un desarrollo de software, es necesario establecer un estándar de codificación para asegurarse de que todos los programadores del proyecto trabajen de forma coordinada. Para un programador comprender bien un sistema de software, influye directamente la legibilidad del código fuente. La mantenibilidad del código es la facilidad con que el sistema de software puede modificarse para añadirle nuevas características, modificar las ya existentes, depurar errores, o mejorar el rendimiento (Larman, 1999).

En la solución propuesta se tendrá en cuenta los siguientes estándares de codificación limitándose al uso de los estándares utilizados por Symfony2 (Figura 12), los cuales están, según (Potencier, y otros, 2008), definidos en los documentos PSR-0, PSR-1, PSR-2.

1. Los archivos solo deben utilizar una codificación UTF-8.
2. Los nombres de las variables, métodos y clases deben expresar el propósito de dicho elemento.
3. Los nombres de clases deben ser escritas utilizando la técnica *StudlyCaps*.
4. La indentación debe ser con un tabulador establecido a 4 espacios.
5. Se escribe únicamente una declaración por línea.
6. El número de caracteres por línea deben ser de 80 columnas, aunque también está aceptado que sean hasta 120.

Se utilizó el estilo *lowerCamelCase* para la creación de los atributos y variables y el estilo *UpperCamelCase* para los métodos y clases.

- Declarar las propiedades de clase antes que los métodos.
- Utilizar mayúsculas intercaladas sin guiones bajos, en nombres de variable, función, método o argumentos.
- El nombre de las variables siempre comienza con el caracter especial „\$“, sin espacio y escrito en minúsculas. En caso de ser un nombre compuesto por más de una palabra, cada una debe escribirse en minúscula, sin espacio y sin guiones.
- Declarar los nombres de las tablas en la base de datos utilizando como separador el guión bajo, en caso de que el nombre sea compuesto.

```
namespace BlogBundle\Controller;
use Symfony\Bundle\FrameworkBundle\Controller\Controller;
use Sensio\Bundle\FrameworkExtraBundle\Configuration\Route;
class DefaultController extends Controller
{
    /**
     * @Route("/")
     */
    public function indexAction()
    {
        return $this->render('BlogBundle:Default:index.html.twig');
    }
    /**
     * @Route("/comments/{id}/toggle", name="comment_toggle_approved")
     */
    public function toggleApprovedAction($id)
    {
        $em = $this->getDoctrine()->getManager();
        $entity = $em->getRepository('BlogBundle:Comment')->find($id);
        $entity->setApproved($entity->getApproved() ? false : true);
        $em->persist($entity);
        $em->flush();
        return $this->redirect($this->get('request')->server->get('HTTP_REFERER'));
    }
}
```

Figura 12 Utilización de los estándares de codificación

4.3 Tratamiento de errores

4.3.1 Tratamiento de errores del lado del cliente

El *framework JQuery* utilizado para la generación de las vistas permite la validación mediante JavaScript de los recursos manipulados, de esta manera se pudieron garantizar las siguientes validaciones:

- Campos en blanco o formato incorrecto: Para cada campo obligatorio se utilizaron validaciones que le indican al usuario cuando el campo fue dejado vacío. Para los campos con formato específico se definieron máscaras para capturar solamente los formatos permitidos (fecha, email). (Ver Figura 13)

Nombre

Por favor introduce un valor

Figura 13 Campos en blanco

- Listas y auto completamiento: La mayoría de las opciones se capturan mediante listas desplegables para evitar la introducción de datos erróneos. En algunos casos, el sistema permite introducir valores y la aplicación se encarga de filtrar la lista y valida siempre que los caracteres introducidos se encuentren dentro del conjunto de datos a seleccionar. (Ver Figura 14)



Figura 14 Autocompletamiento

4.3.2 Tratamiento de errores del lado del servidor

En las clases controladoras se realiza para toda acción las validaciones siguientes:

- Peticiones seguras ejecutadas por el método *Post*: Todas las peticiones que salvan información en la base de datos se ejecutan por el método *Post*, lo que evita el acceso indebido a funciones desde la barra de direcciones del navegador.
- Comprobación de la integridad de datos: Para las acciones de inserción y actualización se valida que no se entren valores iguales a otros registrados en la base de datos, que existan los identificadores y que se procesen las operaciones mediante transiciones.
- Captura de errores de respuesta: Los errores generados por el modelo son enmascarados para transmitir al usuario mensajes que le permitan identificar la causa del error.
- Almacenamiento en secciones de valores de usuario: Los valores fijos de los usuarios conectados se almacenan en variables de sesión, teniéndose control de cada ejecución del usuario y validándose que se tengan los permisos para la acción que se solicita.

4.4 Pruebas

La realización de las pruebas es una de las etapas principales durante todo el ciclo de vida del software. Las mismas constituyen una serie de actividades donde cada componente, o parte de la aplicación a comprobar, son ejecutados bajo determinadas condiciones. Su objetivo principal es evaluar y elevar la calidad del producto final.

Capítulo 4. Implementación y Pruebas

4.4.1 Pruebas de caja negra

La prueba de caja negra se refiere a las pruebas que se llevan a cabo en la interfaz del software. Una prueba de caja negra examina algunos aspectos fundamentales de un sistema con poca preocupación por la estructura lógica interna del software. (Pressman, 2008)

Conocida también como pruebas de comportamiento, se centran principalmente en los requisitos funcionales detectados con anterioridad. Se especializa en la comprobación de interfaces y cuenta con el apoyo de casos de prueba de cada funcionalidad. Su objetivo principal es detectar errores de las siguientes categorías:

- Errores de funciones incorrectas o ausentes.
- Errores de interfaz.
- Errores en estructura de datos o en accesos de datos externas.
- Errores de rendimiento.
- Errores de inicialización y de terminación.

Dentro de los métodos de caja negra se utilizó la técnica de partición de equivalencia. Esta técnica divide el campo de entrada en clases de datos que tienden a ejercitar determinadas funciones del software. Se realizó un caso de prueba por cada caso de uso del sistema. En las Tablas 5, 6, 7, 8, 9 y 10 que se muestran en el Anexo 3, se puede observar el caso de prueba correspondiente al caso de uso Gestionar Evento. Los restantes casos de pruebas se pueden consultar en el Expediente de Proyecto.

Resultados de las pruebas

Durante el transcurso de las pruebas al sistema se detectaron No Conformidades (NC), clasificadas según su importancia en significativas y no significativas. Entiéndase por significativa aquellas NC que puedan afectar el funcionamiento del sistema, no significativas las enfocadas en el diseño u otro aspecto que no afecte el funcionamiento de la propuesta de solución como validaciones y errores de ortografía. A continuación, se muestra el resumen de las No Conformidades detectadas.

Como se muestra en la Figura 15, en la primera iteración se identificaron un total de 17 no conformidades, 3 significativas y 14 no significativas. Luego de realizada las correcciones a las no conformidades detectadas se procedió a una segunda iteración. Se detectaron un total de 13 no conformidades, 4 de ellas clasificadas

Capítulo 4. Implementación y Pruebas

como significativas y 9 no significativas. Finalmente, para una tercera iteración no fueron identificadas ninguna de ellas, lo que confirmó el buen funcionamiento del sistema.

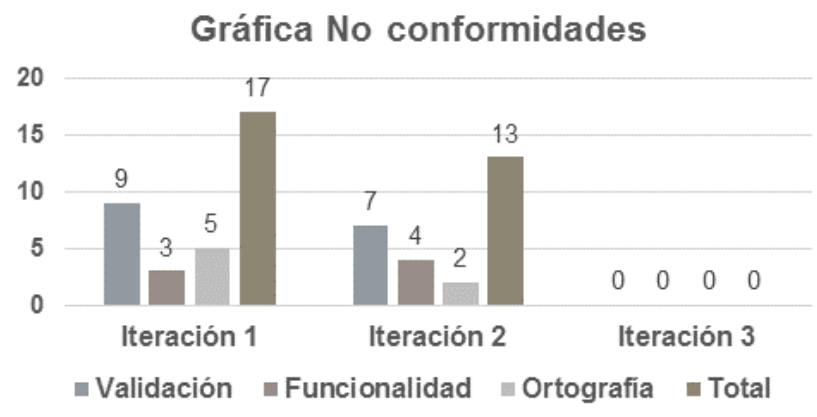


Figura 15 Gráfica de No Conformidades

4.4.2 Pruebas de rendimiento

Las pruebas de rendimiento tienen que diseñarse para asegurar que el sistema pueda procesar su carga esperada. Esto normalmente implica planificar un grupo de pruebas en la que la carga se va incrementado regularmente hasta que el rendimiento del sistema se hace inaceptable. Se ocupan tanto de demostrar que el sistema satisface los requerimientos como de descubrir defectos y problemas en el sistema (Sommerville, 2005).

Para este tipo de pruebas se utilizó la herramienta JMeter. Las pruebas se desarrollaron utilizando una computadora de uso personal Compaq, con 2 Gb de RAM y un procesador Core2Duo con una frecuencia de 2.6 GHz. A continuación, se muestran las variables analizadas:

- Muestra: Cantidad de peticiones realizadas para cada URL.
- Media: Tiempo promedio en milisegundos en el que se obtienen los resultados.
- Mediana: Tiempo en milisegundos en el que se obtuvo el resultado que ocupa la posición central.
- Min: Tiempo mínimo que demora un hilo en acceder a una página.
- Max: Tiempo máximo que demora un hilo en acceder a una página.
- Línea 90 %: Máximo tiempo utilizado por el 90 % de la muestra.

Capítulo 4. Implementación y Pruebas

- Porcentaje de error (%Error): Porcentaje de error de las páginas que no se llegaron a cargar de manera satisfactoria.
- Rendimiento (Rend): El rendimiento se mide en cantidad de solicitudes por segundo.
- Kb/Seg: El rendimiento se mide en cantidad de kilobytes por segundo.

Tabla 4 Prueba de rendimiento al SGICICE

Usuarios	Muestra	Media	Mediana	Línea 90 %	Min	Max	%Error	Rend.	Kb/s
200	1900	1130	1320	1276	2	1288	0	83.4	740.7
400	40900	2243	2609	2046	3	423547	0.17	74.2	662.1
800	52200	2950	5125	7946	4	500956	0.32	66.1	548.9

Las pruebas que se realizaron muestran que el sistema es capaz de responder a 1900 peticiones de 200 usuarios conectados simultáneamente en un tiempo promedio de 1130 milisegundos (1.1 segundos aproximadamente) con 0 % de error.

Por otra parte, se realizaron 40900 peticiones iniciadas por 400 usuarios y en este caso el sistema respondió en 2243 milisegundos (2.2 segundos aproximadamente) como tiempo promedio. No fue capaz de responder correctamente el 0.17 % de las peticiones realizadas.

Por último, se realizaron 51200 peticiones iniciadas por 800 usuarios y en este caso el sistema respondió en 2950 milisegundos (3.0 segundos aproximadamente) como tiempo promedio. No fue capaz de responder correctamente el 0.32 % de las peticiones realizadas.

4.4.3 Valoración de la contribución lograda

A partir de la caracterización del proceso de gestión de la información asociada a los servicios que se brindan en centros que investigan en el área de las Ciencias de la Educación Superior; los autores de esta investigación concluyen que: *la utilización del sistema de gestión de la información del Centro de Innovación y Calidad de la Educación, permite valorar como positiva la contribución a los grados centralidad, disponibilidad, seguridad e integridad de la información que se gestiona a través de la aplicación anteriormente mencionada.*

Capítulo 4. Implementación y Pruebas

Se arriba a dicha conclusión, teniendo en cuenta que:

- El resultado de las pruebas de caja negra realizadas mediante los casos de prueba, evidencia que la información obtenida del sistema aumentó de manera significativa los grados de completitud, corrección y precisión de los datos, lo cual representa una contribución al grado de integridad de la información en el PGICICE-UCI.
- Con las pruebas de rendimiento aplicadas se evidenció la alta capacidad de respuesta de la aplicación ante las diversas peticiones de acceso a información al mismo tiempo. Teniendo en cuenta que la disponibilidad de la información depende de la capacidad de respuesta de su propia fuente, se infiere un alto grado de disponibilidad a la información.
- Se evidencia que toda la información analizada como parte de esta investigación se encuentra centralizada, y se puede acceder a la misma de forma rápida y organizada a través de una aplicación que pertenece al CICE-UCI.
- A partir de las pruebas de caja negra que se realizaron y la propia utilización del *framework* Symfony con el ORM Doctrine se garantiza la seguridad de la aplicación. Esto es posible a partir de la gestión de usuarios implementada, la definición de roles y permisos y la autenticación en el sistema. Doctrine, que es el ORM que utiliza Symfony para la conexión con la base de datos, se encarga de prevenir las inyecciones SQL. De igual forma se definen las rutas a las que tendrá acceso cada usuario. Si existe una persona que desea quebrantar la seguridad, introduciendo directamente en el navegador una ruta a la cual no debe acceder, el sistema muestra de manera automática un formulario para que el usuario se autentique.

La aplicación desarrollada cumple con los principios anteriormente mencionados, directamente asociados a las variables *centralidad*, *disponibilidad*, *seguridad* e *integridad*, las cuales estaban limitadas sustancialmente.

Capítulo 4. Implementación y Pruebas

Conclusiones parciales

A partir del desarrollo del presente capítulo se arribaron a las siguientes conclusiones:

- Se detalló el tratamiento de errores de la aplicación donde se evidencia la capacidad del sistema de responder ante situaciones no esperadas.
- Los estándares de codificación y los estilos de programación utilizados fueron descritos para hacer más fácil el entendimiento del código por el desarrollador, así como un mejor mantenimiento a la aplicación en el futuro.
- Se realizaron las pruebas al sistema donde se emplea el método de caja negra y se aplica la técnica de partición de equivalencia. Esto permitió verificar los requisitos funcionales del sistema, demostrar que las funciones de la aplicación son operativas, producen un resultado satisfactorio y el sistema cumple con los objetivos trazados.

Conclusiones

- Se automatizó el proceso de gestión de información del Centro de Innovación y Calidad de la Educación ya que presentaba bajos grados de centralidad, disponibilidad, integridad y seguridad. Se construyó un sistema con un enfoque de aplicación web siguiendo la metodología OpenUP, utilizando el *framework* Symfony 2.8 y el SGBD PostgreSQL 9.4.5.
- A partir de la aplicación de técnicas de obtención de información se definieron los requisitos funcionales y no funcionales del sistema, para ofrecer las funcionalidades y capacidades. El lenguaje UML permitió modelar el sistema a desarrollar, el modelo lógico y físico de los datos; así como información solicitada por el cliente para facilitar el futuro mantenimiento y evolución de la aplicación.
- Para comprobar la calidad y correcto funcionamiento del sistema, se diseñaron y ejecutaron casos de prueba, los que arrojaron resultados satisfactorios, lo cual demuestra el cumplimiento de los requisitos funcionales establecidos en la fase inicial del proceso de desarrollo del software.
- Se cumplió el objetivo general de la investigación al desarrollar el sistema de gestión de información del CICE, como resultado de la investigación realizada. Este será de gran utilidad ya que permite el acceso a los servicios que se brindan en él y logran altos grados de centralidad, disponibilidad, seguridad e integridad de la información que allí se maneja.

Recomendaciones

Adicionar nuevas funcionalidades que permitan gestionar información referente a otros servicios que se realizan en el CICE como son:

1. Ciencia, Tecnología e Innovación: Adicionar por cada perfil los temas de doctorado, maestría y estudiantes que tutora, así como su vinculación a un proyecto de investigación. De igual forma agregar los premios de Ciencia y Técnica obtenidos, así como resultados introducidos en la práctica.
2. Organización y dirección de la comisión científica.

Bibliografía

1. **Abrahamsson, P y otros. 2002.** *Agile software development methods- Review and analysis.* Universidad de Oulu. Finlandia : ESPOO, 2002.
2. **Almerira, S y Pérez Cavenago, V. 2007.** *Arquitectura de Software: Estilos y Patrones.* 2007.
3. **Apache Software Foundation. 2000.** *Apache Tomcat 6.0 User Guide.* [En línea] 2000. [Citado el: 23 de 3 de 2014.]
4. **Balduino, R. 2007.** *Introduction to OpenUP (Open Unified Process).* 2007. Disponible en: <http://www.eclipse.org/epf/general/OpenUP.pdf>.
5. **Bartle, P. 2011.** *Información para la gestión.* Disponible en: <http://cec.vcn.bc.ca/mpfc/modules/mon-miss.html>.
6. **Bueno Campos, E. 2001.** *De la sociedad de la información a la del conocimiento: experiencias en España, en CIED: «Gerencia Del conocimiento. Potenciando el Capital Intelectual para crear valor».* 2001.
7. **Bustelo, C y Amarilla, R. 2001.** *Gestión del conocimiento y gestión de la información.* 2001.
8. **Cáceres, M, y otros. 2010.** *La formación pedagógica de los profesores universitarios. Una propuesta en el proceso de profesionalización del docente.* Cuba: s.n., 2010.
9. **Castillo Cantón, A. 2011.** *Manual de HTML5 en español .* 2011.
10. **Capote, B, y otros. 2003.** *La gestión de información como herramienta fundamental en el desarrollo de los centros toxicológicos.* La Habana, Cuba: s.n., 2003
11. **CBASQA. 2008.** *Proceso de Desarrollo OpenUP: BASQA – Desarrollo de Software, SQA, Testing, Servicios Informáticos, Project Management. BASQA – Desarrollo de Software, SQA, Testing, Servicios Informáticos, Project Management.*
12. **Douglas, K and Douglas, S. 2005.** *PostgreSQL: The comprehensive guide to building, programming and administering PostgreSQL databases.* Second Edition. s.l. : Editorial Sams Publishing, 2005. p. Introduction. PostgreSQL features.
13. **Dunglas, K.** *Persistence in PHP with the Doctrine ORM.* s.l. : Packt Publishing Ltd, 2013.
14. **Egulliz, J. 2013.** *Desarrollo web ágil con Symfony 2.3.* 2013.

15. **Espinosa Fuentes, F. 2013.** *Sistema de informacion para la gestion de empresa.* s.l. : Universidad de Talca, 2013.
16. **Fernández, A. 2009.** *Arquitectura de información de los portales intranets: un componente esencial de la gestión de información en las universidades.* 2009.
17. **Figueroa, R y otros. 2011.** *Metodologías Tradicionales Vs. Metodologías Ágiles.* Universidad Técnica Particular de Loja. Escuela de Ciencias en Computación, 2011
18. **Fowler, M, Rice, D y Foemmel, M. 2002.** *Patterns of Enterprise Application Architecture.* s.l. : Addison Wesley, 2002. ISBN : 0-321-12742-0 .
19. **Gadja, W. 2007.** *Instant PhpStorm Starter.* Birmingham : Packt Publishing Ltd, 2007.
20. **González García, A y Ferrer Parés, M. 2012.** *Gestión del Conocimiento en Cuba: diseminación de sus resultados de investigación, de 1997-2010.* 2012.
21. **González Díaz, Y y Fernández Romero, Y. 2012.** *Patrón Modelo-Vista-Controlador.* 2012.
22. **Hernández, R y González, S. 2011.** *El proceso de investigación científica.* La Habana, Cuba: Editorial Universitaria, 2011.
23. **IEEE Std 610.12. 1990.** *IEEE Standard Glossary of Software Engineering Terminology.* New York, USA: s.n., 1990.
24. **Jacobson, I, Booch, G y Rumbaugh, J. 2000.** *El proceso unificado de desarrollo de software.* Madrid : Pearson Educación, 2000. ISBN: 84-7829-036-2.
25. **Larman, C. 1999.** *UML y Patrones. Introduccion al análisis y diseño orientado a objetos.* 1999.
26. **Martínez, R. 2013.** *PostgreSQL-es.* [En línea] 2013. [Citado el: 15 de Abril de 2014.] <http://www.postgresql.org.es.htm>.
27. **MES. 2004.** *Reglamento de la Educación de Posgrado de la República de Cuba.* MES. La Habana, Cuba: s.n., 2004. Resolución Ministerial 132/2004.
28. **Microsoft. 2014.** [En línea] 2014. [Citado el: 22 de Marzo de 2015.]
29. **McGraw-Hill. 2001.** *Diccionario de informática e internet de Microsoft.* s.l.: McGraw- Hill, Interamericana de España, 2001.
30. **Myers, G y Badgett, T. 2011.** *The art of software testing.* s.l. : John Wiley & Sons, 2011.

31. **Nonaka, I y Takeuchi, H. 1999.** *Teoría de la creación del conocimiento organizacional. La dimensión ontológica y la dimensión epistemológica.* 1999.
32. **Núñez Jover, J, Montalvo Félix, L. y Ones Pérez, I. 2006.** *La gestión del conocimiento, la ciencia, la tecnología y la innovación en la nueva universidad: una aproximación conceptual. La nueva universidad cubana y su contribución a la universalización del conocimiento.* La Habana, Cuba. : Félix Varela, 2006.
33. **Okeke Oluchukwu, E.** *What's the best editor/IDE for PHP.* Quora. [En línea] 10 de Octubre de 2015. <https://www.quora.com/Whats-the-best-editor-IDE-for-PHP>.
34. **Paradigm, Visual. 2013.** *Visual paradigm for uml. Visual Paradigm for UML-UML tool for software application development.* 2013.
35. **Penadés, C y Letelier, P.** *Ingeniería del software y sistemas de información.* 2014.
36. **Pérez Váldez, D. 2015** *Los diferentes lenguajes de programación para la web.* <http://www.maestrosdelweb.com/los-diferentes-lenguajes-de-programacion-para-la-web/>.
37. **Potencier, F. y François, Z. 2008.** *Symfony, la guía definitiva.* EE.UU: libroweb, 2008.
38. **PostgreSQL Global Development Group. 2015.** *PostgreSQL 9.4 Documentation.* Berkeley : s.n., 2015.
39. **Pressman , R. 2008.** *Ingeniería de Software: un enfoque práctico.* s.l. : Mc Graw Hill, 2008. ISBN: 970-10-5473-3.
40. **Reynoso, C.** *Introducción al Patrón de Arquitectura por Capas.* Argentina: s.n., 2004. Vol. 33.
41. **Ronda, R. 2004.** *Arquitectura de Información: caminos prácticos,* 2004.
42. **Sampieri, R. y Fernandez, C. 2006.** *Metodología de la Investigación Científica.* México: McGraw-Hill, 2006.
43. **Sommerville, I. 2005.** *Ingeniería del Software.* Madrid, España: Pearson Educación S.A, 2005
44. **Soto Balbón, M. Aurora y Barrios Fernández, N. 2006.** [En línea] 2006. http://bvs.sld.cu/revistas/aci/vol14_2_06/aci04206.htm#cargo.
45. **Tinoco Gómez, O, Rosales Lopez, P. Pablo y Salas Bacalla, J. 2010.** *Criterios de selección de metodologías de desarrollo de software.* 2010.
46. **Wesley, A. 2004.** *Softwre development for small team: A RUP - centric approach.* s.l. : Addison Wesley object technology series, 2004. ISBN: 978-0321-199-50-8.

