

Universidad de las Ciencias Informáticas



Facultad 10

SISTEMA WEB DE INFORMATICA 2009

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

Autor:

Pedro Enrique Castiñeiras Sánchez

Tutor:

Ing. Maikel Fernández Fernández

**Ciudad de La Habana, Cuba
Junio, 2007**

“...de luz se han de hacer, y deben dar luz...”

José Julián Martí Pérez

Agradecimientos

A mi tutor por haberme guiado en la preparación de este trabajo de diploma y por su defensa justa en los momentos que ha hecho falta.

A las profesoras Maidely Calderón Montero y Yanedi Abreu Bartomeo por su ayuda ante mis dudas en la realización de este trabajo, fundamentalmente en los temas de ingeniería de software.

A la Lic. Xiomara Hernández Rojas del MIC por su tiempo dedicado a explicarme los procesos de la feria y la convención.

A mi amigo Rolan Rober Bullain Diéguez por su inmenso apoyo y ayuda, por cumplir función de segundo tutor.

A mi amiga Lourdes Elena Stusser Iglesias por brindarme siempre su apoyo.

A mi amigo Fernando de Jesús Cabrera Almelo por enseñarme a caminar en el mundo informático.

A mi familia toda y en especial a mi tío Roberto Sánchez, mi abuela Modesta, mi papá y mi mamá por mantenerme todos estos años, permitiéndome solo ocuparme de mis estudios. A ustedes que les debo lo que soy, mis triunfos son y serán de ustedes.

A mi novia por apoyarme y ayudarme tanto, por enseñarme RUP y estar siempre a mi lado. Taymí te amo.

A toda mi familia de Tunas por darme siempre unas vacaciones tan buenas y hacerme un pedacito en sus corazones.

A mis hermanos por ser mi impulso y mi vida.

A mi padre y madre por confiar en mí.

A mi padre, amigo y guía por su ejemplo. Te amo pipo.

A mi madre, por su comprensión, apoyo, malcriadeces, cariño y paciencia. Te amo mima.

A todas las personas que de una forma u otra han colaborado con la realización de este trabajo.

A toda mi gente, gracias mil por hacerme sentir apoyado.

Dedicatoria

Para mis padres por hacerme lo que soy

La Convención y Feria Internacional de Informática es el acontecimiento más importante que se desarrolla en Cuba en las ramas informática, telecomunicaciones, electrónica y automática desde el año 1988.

A lo largo de las diferentes ediciones realizadas el evento ha ido ganando una amplia participación nacional e internacional. En la convocatoria próxima a este estudio se estima que la cantidad de solicitudes para participar resultará considerablemente grande.

El principal designio de los anfitriones es propiciar un extenso intercambio científico, en aras del desarrollo, pero las condiciones actuales no les permiten tales propósitos en la escala deseada, por lo que se da la necesidad de realizar el presente trabajo, que constituye un trabajo de diploma, con el objetivo de realizar el análisis y diseño del sistema Web para Informática 2009, que sirva para el posterior desarrollo de dicho sistema, el cual permita la participación a todas las personas interesadas que no puedan asistir presencialmente.

En este estudio se analizan las metodologías de desarrollo, tecnologías, técnicas, lenguajes y herramientas más utilizadas, a partir de las cuales se hace una propuesta sobre cuales deben ser las empleadas en la realización del producto. Se realiza la fase de modelación del negocio de la metodología RUP, y posteriormente se realiza la fase siguiente de análisis y diseño del sistema a partir de la captura de requisitos.

The Informatics International Convention and Fair is the most important event that takes place in Cuba regarding to informatics, telecommunications, electronics and automatic since 1988.

The participation in the event has increased widely, not only nationally but also internationally, throughout the various editions made. In the next call to this study is estimated that the number of participation request will be considerably larger.

The main intend of the hosts is to facilitate an extensive scientific exchange, for the sake of development, but current conditions do not allow them to fulfill such purposes in the desired scale, so it is justify the development of this research, which main goal is to accomplish the analysis and design of Informatics 2009 Web system and that constitutes a degree thesis, to the further development of this system, which will allow the participation of all interested persons who can not be involved personally.

This study analyses the development methodologies, technologies, techniques, tools and languages most commonly used, from which there is a proposal which must be employed to carrying out the product. We carried out the phase of the business modeling methodology according RUP, and then perform the next stage of analysis and system design based on the client requirement.

Introducción	1
Capítulo I. Fundamentación teórica	5
1.1 Estado del arte	5
1.2 Las aplicaciones Web.....	6
1.3 Metodologías de desarrollo.....	7
1.3.1 El proceso unificado de desarrollo (RUP).....	8
1.4 Arquitectura de software.....	9
1.4.1 Patrones arquitectónicos.....	10
1.5 Diseño de software.....	11
1.5.1 Patrones de diseño.....	11
1.6 Herramientas de desarrollo.....	12
1.6.1 Servidores de aplicaciones Web.....	13
1.6.2 Entornos de desarrollo	14
1.6.3 Software cliente del Sistema Gestor de Base de Datos.....	15
1.6.4 Herramientas Case	15
1.6.5 Controlador de versiones.....	16
1.6.6 Navegadores de Internet.....	16
1.7 Lenguajes.....	17
1.7.1 Lenguaje de programación PHP.....	17
1.7.2 Lenguaje de modelado unificado.....	18
1.8 Sistemas manejadores de contenidos.....	20
1.8.1 ¿Qué CMS elegir?.....	20
1.9 Los sistemas gestores de bases de datos.....	21
Capítulo II. Caracterización del sistema.....	24
2.1 Objeto de estudio.....	24
2.1.1 Objeto de automatización.....	24
2.1.2 Información que se maneja.....	24
2.1.3 Propuesta de sistema.....	25
2.1.4 Modelo de negocio.....	25
2.1.5 Breve descripción del negocio.....	26
2.1.6 Diagrama de casos de uso del negocio.....	27
2.1.7 Diagramas de actividades.....	29
2.2 Especificación de los requisitos de software.....	34
2.2.1 Requerimientos no funcionales.....	34
2.2.2 Requerimientos funcionales.....	40
2.3 Modelo de caso de uso del sistema.....	42
2.3.1 Definición de los actores.....	42
2.3.2 Listado de casos de uso.....	43
2.3.3 Diagrama de casos de uso del sistema.....	47
2.3.4 Casos de uso expandidos.....	51
Capítulo III. Análisis y diseño del sistema.....	71
3.1 Modelo de análisis.....	71
3.1.1 Diagrama de clases del análisis.....	71
3.2 Modelo de diseño.....	72
3.2.1 Diagrama de clases con estereotipos Web.....	72

3.2.2 Diagramas de interacción.....	81
3.2.3 Diseño de la BD	81
Conclusiones.....	83
Recomendaciones.....	84
Anexos	85
Anexo 1. Tabla comparativa entre CMS.....	85
Anexo2. Diagramas de clases.	87
Anexo3. Diagramas de interacción.	92
Anexo 4. Diagramas de clases de diseño	100
Anexo 5. Diagrama entidad relación.....	105
Anexo 6. Paquete BD. Diagrama de clases persistente.....	107
Glosario de términos	108
Bibliografía	115

Introducción.

La Convención y Feria Internacional de Informática es el evento más importante de esta rama que se desarrolla en Cuba desde el año 1988 con una periodicidad de dos años. Es un acontecimiento de carácter internacional, en el que participan instituciones, universidades, centros de investigación de Ibero América y todo aquel público en general que esté interesado. Este importante acontecimiento está estructuralmente dividido en la Feria y la Convención.

En la feria se exponen los más novedosos proyectos, tecnologías, recursos, productos y servicios del ámbito informático, telecomunicaciones, electrónica y automática por productores de empresas cubanas y extranjeras en un clima de intercambio y solidaridad; y se realizan encuentros entre inversionistas de los distintos sectores.

En la convención se realizan un conjunto de congresos, seminarios, conferencias magistrales y talleres con un enfoque científico, que reúnen lo más novedoso de todos los trabajos que aspiran a ser presentados en cada cita, los cuales pueden ser aprobados o no, según la calidad que presenten, calificados por tribunales designados para este fin, desde fechas cercanas a la realización de este acontecimiento.

Con el paso del tiempo, este evento ha ganado un gran realce en la comunidad científica internacional, y la cantidad de trabajos candidatos para la convención y solicitudes de stand expositivo para la feria en cada oportunidad son mayores, llegando a cifras considerablemente complicadas para planificar su exposición en lo que respecta al tiempo y la logística necesaria.

Ante esta circunstancia surge la necesidad del presente trabajo, diseñar un sistema que colabore con la Convención y Feria Internacional de Informática 2009, brindando la posibilidad de acoger la mayor cantidad posible de participantes que no se vean limitados a presentarse físicamente en el evento y puedan disfrutar de sus beneficios virtualmente, en los salones virtuales de eventos, las relatorías del evento y las galerías de fotos como paseos virtuales por la feria. De esta manera todos los interesados en participar en el evento puedan ser gratificados con los beneficios que el mismo brinda.

Al analizarse tal **situación problemática** se determinó que el **problema** ante el que se encuentra este estudio es ¿Cómo diseñar una herramienta que colabore con el evento Informática 2009 que permita obtener una mayor participación de delegados y espectadores en este acontecimiento?

El proceso de desarrollo de software constituye el **objeto de estudio** de este trabajo de tesis puesto que es la ciencia que estudia el fenómeno y se ha delimitado como **campo de acción** el diseño de aplicaciones Web utilizando sistemas manejadores de contenidos (CMS, content management system) bajo licencias no privativas.

En esta investigación se defiende la **idea** siguiente: la propuesta de análisis y diseño de un sistema virtual para la Convención y Feria Internacional de Informática 2009, permitirá a los desarrolladores implementar el sistema que gestione y controle los procesos de tal evento.

Este trabajo tiene como **objetivo general** elaborar el análisis y diseño de una aplicación Web para la Convención y Feria Internacional de Informática a celebrarse en el año 2009. Dicha meta puede ser lograda con la culminación de los **objetivos específicos** siguientes:

- Comprender los procesos que se desarrollan en el evento.
- Realizar un estudio del estado del arte.
- Investigar sobre las aplicaciones distribuidas, estableciendo comparaciones sobre las distintas tecnologías, técnicas y lenguajes.
- Modelar los procesos de negocio
- Elaborar el análisis y diseño a partir de los discernimientos logrados en el cumplimiento de los objetivos anteriores.

Para lograr los objetivos antes mencionados este estudio se propone cumplir un grupo de **tareas investigativas** que servirán de guía a este trabajo de diploma, estas son:

- Analizar todos los procesos que ocurren en el evento para lograr un entendimiento total del mismo.
- Estudiar las tendencias de aplicaciones distribuidas, para resolver problemáticas similares en otros eventos de carácter internacional.
- Examinar las peculiaridades de los sistemas gestores de contenidos (CMS), y determinar el más adecuado.
- Analizar las distintas tecnologías Web para concluir cuales se usarán como parte de la solución.
- Determinar las herramientas a utilizar para desarrollar el proyecto.
- Modelar todos los procesos del negocio que serán objeto de automatización.
- Confeccionar el análisis y diseño del sistema Web concebido con los conocimientos obtenidos en las anteriores tareas.

Este trabajo de diploma está compuesto por tres capítulos. El **primer capítulo**, titulado "Fundamentación teórica", muestra el estudio realizado del estado del arte y la fundamentación teórica, analizando las diferentes estrategias de solución que se han elaborado para otros eventos

internacionales de igual demanda, se examinan las distintas metodologías para el desarrollo de software, las tecnologías más notorias en las aplicaciones distribuidas, las herramientas necesarias para el desarrollo de este proyecto, los lenguajes de programación más utilizados en la concepción de sistemas Web con prestaciones similares a las necesarias, el uso de sistemas manejadores de contenidos (CMS), se profundiza sobre los diferentes patrones arquitectónicos y de diseño, y las ventajas que aportan estos a la solución propuesta.

Bajo el nombre “Características del sistema”, el **segundo capítulo** desarrolla al flujo de trabajo modelación del negocio y captura de requisitos funcionales y no funcionales, se dan a conocer las características del sistema en general, se expone el problema a resolver y la solución propuesta para erradicar dichos problemas. Se plantean los procesos que son objeto de automatización y se describe la información fundamental que se maneja. Se realiza la descripción general de los procesos de negocio identificados, actores del negocio, actores del sistema, trabajadores del negocio, casos de uso, diagramas de casos de uso del negocio, diagramas de actividades, modelos de objetos, diagrama de casos de uso del sistema, la definición de los actores y los casos de uso expandidos.

El **tercer capítulo** lleva por título “Análisis y diseño del sistema” y se hace referencia al flujo de trabajo análisis y diseño de la metodología RUP, en este se definen el modelo de análisis y el diagrama de clases del diseño con los diagramas de interacción correspondientes a los casos de uso arquitectónicamente significativos y sus diferentes escenarios. Además se incorpora el modelo entidad relación de los datos.

Capítulo I. Fundamentación teórica.

Toda investigación requiere un basamento teórico, mediante el cual se realice un estudio crítico y valorativo de los diferentes temas a tratar.

En este capítulo, se comienza realizando un estudio del estado del arte, donde se analizan las diferentes tendencias a nivel nacional e internacional sobre el tema investigado. Posteriormente, se reproducen ciertas informaciones estadísticas, a través de las cuales se establecen comparaciones y juicios sobre las diferentes metodologías de desarrollo, tecnologías, lenguajes de programación y herramientas, preferentemente bajo licencias no privativas, como defensores del software libre.

1.1 Estado del arte.

Los sistemas Web tienen en la actualidad una gran aceptación en los cibernautas de Internet, debido a las grandes oportunidades que brindan este tipo de aplicaciones, como lo son: la simple necesidad, por parte de los clientes, de un navegador Web para hacer uso de los sistemas; la disponibilidad de estos servicios en cualquier punto de acceso a la red en que se encuentre alojado y su fácil forma de actualización. Estas características han sido aprovechadas por empresas e instituciones para dar a conocer productos y servicios, o cualquier otro beneficio para las mismas, y no están exentas de este grupo las entidades científicas, que en aras de promover sus intereses se apoyan en sistemas distribuidos, tal es el caso de organizaciones científicas que organizan eventos internacionales y promueven la participación en estos a través de sistemas Web, mediante las cuales, en el mejor de los casos, los usuarios de la red de redes (Internet) pueden participar virtualmente usando su computadora personal desde diversas partes del mundo, presentando trabajos, viendo los trabajos presentados, participando en foros con los innovadores presentes en tales acontecimientos, así como otras peculiaridades de reuniones de este tipo.

Entre las instituciones que han desarrollado sistemas como soporte a los eventos internacionales que realizan se encuentran: el Instituto de Empresa Business School en España ((IEBS) 2005), PDV Merchandising SA en Argentina (MERCHANDISING), SIMO Feria Internacional de Informática, Multimedia y Comunicaciones en España (IFEMA), los Encuentros Académicos Internacionales (EUMED), el Congreso Internacional Virtual de Educación (CIVE), entre otros; pero no fue posible precisar los detalles de estas soluciones puesto que es necesario abonar pagos para poder disfrutar de los mismos, aunque si se pudo apreciar la concepción e idea básica de estos.

El Ministerio de Informática y las Comunicaciones de Cuba (MIC), quien se encarga de la organización de la Convención y Feria Internacional de Informática con cita en la mayor de las Antillas, en la edición

Capítulo I. Fundamentación teórica.

del año 2007 presentó un sistema para el evento virtual ((MIC) 2007), pero que no logró satisfacer todas las expectativas. Este trabajo se realiza en coordinación con el MIC, con el fin de ser implantado en el próximo encuentro.

1.2 Las aplicaciones Web.

Un sistema o aplicación Web es un producto informático basado en la arquitectura Cliente-Servidor que ha permitido un flujo de comunicación global a una escala sin precedentes en la historia humana, que los usuarios utilizan a través de Internet o de una intranet. Son muy populares debido a que son programas informáticos muy fáciles de actualizar, ya que solo requieren la actualización de los servidores, son además aplicaciones que pueden ser accedidas desde cualquier lugar, con la sola utilización de un navegador Web, permitiendo acceder a numerosos servicios de la red, no ocurriendo así con aplicaciones de escritorio.

Como también es conocida, la World Wide Web (WWW o simplemente la "Web") es un entorno donde existen documentos de hipertexto enlazados y accesibles a través de Internet. Mediante el uso de un navegador Web, o también conocidos como explorador Web, un usuario visualiza páginas Web que pueden contener texto, imágenes, videos u otros contenidos multimedia, y navega a través de ellas usando enlaces.

Las aplicaciones Web atienden peticiones utilizando el Protocolo de Transferencia de Hiper-Texto (HTTP, Hypertext Transfer Protocol) que realizan los clientes a través de sus navegadores escribiendo el Localizador Uniforme de Recurso (URL, Uniform Resource Locator) o siguiendo un enlace de hipertexto, procesan la información que se requiera y generan en forma de respuestas, páginas Web en un formato estándar, definido por el World Wide Web Consortium (W3C), consorcio internacional a cargo de los estándares de la Web, denominado Lenguaje de Etiquetas de Hipertexto (HTML, HyperText Markup Language). El HTML le da estructura a la información, y es mediante las hojas de estilo en cascada (CSS, Cascading Style Sheets) que se le da forma a estos datos, así como se brinda interacción con el apoyo de lenguajes script del lado del cliente como JavaScript o Visual Basic Script, combinación denominada HTML Dinámico (DHTML, Dynamic HTML), a la cual la W3C define como un "término utilizado por algunos distribuidores informáticos para describir la combinación de HTML, CSS y lenguajes scripts que permite brindar interacción en los documentos" (HÉGARET 2008).

Este tipo de software, las aplicaciones Web, presentan la desventaja que sus interfaces tienen ciertas limitaciones de funcionalidad, como dibujar en la pantalla, arrastrar y soltar, y otros tipos de interacciones, que en muchos casos los desarrolladores tienen que auxiliarse de lenguajes script del lado del cliente para lograr tal fin. Otra desventaja la constituye que no es posible mediante HTML

Capítulo I. Fundamentación teórica.

enviar y recibir datos de los servidores sin recargar las páginas completamente, para lo que surge un nuevo recurso que lo constituye la tecnología AJAX, Javascript Asíncrono y XML, que no es más que aplicaciones script que se ejecutan en los navegadores Web, colaborando con los servidores de aplicaciones en la transferencia de información sin necesidad de realizar una recarga total a las páginas.

Estos tipo de software no presentan inconvenientes con respecto a los sistemas operativos en las computadoras de los usuarios, debido a que los navegadores Web solo deben interpretar los estándares predefinidos por el W3C, en texto plano, y no ejecutar instrucciones de software, con la excepción de los Applets, mientras que las aplicaciones de escritorio se crean para plataformas específicas, con la excepción de las aplicaciones que utilizan máquinas virtuales para su ejecución, las cuales requieren más recursos que los necesarios por las aplicaciones Web.

Las aplicaciones Web son conocidas también como sistemas distribuidos, ya que los productos que de estas se obtienen generalmente no se generan de un ordenador aislado, sino de la unión de procesos llevados a cabo por diferentes computadoras con un objetivo común, ganando en todo momento en rendimiento, fiabilidad, escalabilidad y efectividad de los costes.

Debido a que el producto es necesario publicarlo en Internet, para que puedan acceder a el todos los cibernautas sin necesidad de tener instalado sistemas en los ordenadores locales, y las ventajas que aportan las aplicaciones Web este será el tipo de aplicación a diseñar.

1.3 Metodologías de desarrollo.

Para lograr el desarrollo de software satisfactoriamente en proyectos de gran tamaño es necesario trabajar bajo metodologías de desarrollo, ya que este es un proceso riesgoso y en ocasiones difícil de controlar. Suele ocurrir que en los proyectos que no se establece una metodología a utilizar, o se emplea mal dicha metodología, lo que se obtiene son clientes y desarrolladores insatisfechos con los resultados. Muchas veces no se elige una metodología adecuada a las condiciones del proyecto, sobre todo cuando se trata de proyectos pequeños que requieran poco tiempo. En estos casos lo que se hace es separar rápidamente el objetivo final en procesos, cada proceso en funciones, y por cada función determinar un tiempo aproximado de desarrollo.

En los proyectos de mayor amplitud, tiene más sentido basarse en una metodología de desarrollo, y buscar cual es la más propicia según las especificidades del proyecto. Muchas veces no se encuentra la más adecuada y se termina por crear una metodología propia, algo que no esta mal, siempre que se logre el objetivo final del proyecto.

Capítulo I. Fundamentación teórica.

Establecer una metodología como técnica de desarrollo puede proporcionar al sistema de flexibilidad para los cambios que solicite el usuario una vez el producto esté terminado o para la evolución del sistema, contribuyendo así a que no se produzcan atrasos en el proyecto por cuestiones organizativas, entendimiento y comunicación entre los miembros del equipo de desarrollo y por tanto mitiga incomodidad ante los cambios en los desarrolladores y clientes. Razón por la cual las metodologías establecen realizar acuerdos formales con los clientes, de tal manera que cada cambio o modificación no perjudique al desarrollo del mismo.

Existen diversas metodologías de desarrollo, cada una con sus peculiaridades, ventajas y desventajas. Tres de las metodologías más utilizadas son el Proceso Unificado de Desarrollo (RUP, Rational Unified Process), Programación Extrema (XP, Extreme Programming) y el Framework de Soluciones de Microsoft (MSF, Microsoft Solutions Framework). RUP es una metodología muy detallada, más adaptable para proyectos de largo plazo, contrario a XP que está recomendada para proyectos de más premura, ya que incluye procesos más ágiles como por ejemplo que los clientes o usuarios se convierten en miembros del equipo de desarrollo, razón por la cual no se puede utilizar esta metodología, por su parte MSF se adapta a proyectos de cualquier dimensión, pero propone la utilización de herramientas de Microsoft, punto que no es afín con este proyecto por las licencias de estos (SANCHEZ).

En este estudio se adoptará RUP como metodología de desarrollo, debido a que la dimensión del proyecto es muy amplia, y se precisa además desarrollar bajo una metodología bien detallada en documentación, para garantizar la evolución del producto en las diferentes versiones sin necesidad de recomenzar el mismo.

1.3.1 El proceso unificado de desarrollo (RUP).

La metodología RUP, divide en cuatro fases el desarrollo del software:

- Inicio: En esta etapa se determina la visión del proyecto.
- Elaboración: Fase donde se obtiene la arquitectura.
- Construcción: Se logra la capacidad operacional inicial.
- Transmisión: Se obtiene una versión entregable del sistema.

Las fases se logran mediante el ciclo de iteraciones. Los Objetivos de cada iteración se establecen en función de la evaluación de las precedentes, donde cada iteración en su desarrollo produce artefactos de entrada a la siguiente. Al finalizar las cuatro etapas se obtiene una versión entregable del sistema que es entregado a los clientes.

Capítulo I. Fundamentación teórica.

Esta metodología está dirigida por los Casos de Uso, centrada en la arquitectura, y es iterativa e incremental.

En el ciclo de vida de cada iteración se realizan una serie de tareas que se pueden dividir en dos disciplinas, desarrollo y soporte, las cuales quedan estructuradas como sigue:

Disciplina de Desarrollo

- Modelado del negocio: Entendiendo las necesidades del negocio.
- Requerimientos: Capturando las necesidades de los clientes y modelar el sistema automatizado que supla dichas necesidades.
- Análisis y Diseño: Traslado de los requerimientos dentro de la arquitectura de software.
- Implementación: Creando software que se ajuste a los modelos establecidos en el análisis y diseño.
- Pruebas: Asegurando que el producto cumple las expectativas de los clientes.

Disciplina de Soporte

- Despliegue: Asegurar que el producto final este listo para entregar a los usuarios finales.
- Configuración y administración cambios: Guardando todas las versiones del proyecto. Así como gestionar las distintas configuraciones.
- Administrando el proyecto: Administrando y planificando tiempo y recursos.
- Ambiente: Brindando soporte al proyecto con las adecuadas herramientas, procesos y métodos.

1.4 Arquitectura de software.

Existen varias definiciones de Arquitectura del Software, pero en general todas giran sobre la idea de que la Arquitectura del Software es el diseño de más alto nivel de la estructura de un sistema, programa o aplicación y tiene la responsabilidad de definir los módulos principales, las responsabilidades de cada uno, y sus interacciones, así como define el control y flujo de los datos, la secuenciación de la información, protocolos de comunicación y ubicación de todas las partes en el hardware.

La IEEE define la Arquitectura del Software en el estándar Std 1471-2000, como: "la organización fundamental de un sistema formada por sus componentes, las relaciones entre ellos y el contexto en el que se implantarán, y los principios que orientan su diseño y evolución" (IEEE-SA 2004).

Capítulo I. Fundamentación teórica.

1.4.1 Patrones arquitectónicos.

Los patrones arquitectónicos son soluciones arquitectónicas para problemas comunes de este tipo, proponen métodos de estructuración de los sistemas.

Existen diversos estilos arquitectónicos expresados mediante patrones, como lo son el patrón modelo-vista-controlador, la arquitectura en capas, la arquitectura basada en componentes, la arquitectura orientada a servicios, entre otros.

El patrón Modelo Vista Controlador (MVC) presenta algunos beneficios, como lo son: proporciona bajo acoplamiento, ya que se desacopla la forma en que se muestran e introducen los datos de los modelos; brinda mayor cohesión puesto que cada elemento del patrón se especializa en su tarea (la vista muestra los datos, el modelo se encarga de la lógica de negocio y el controlador de las entradas); las vistas proveen mayor flexibilidad en la interacción con los usuarios, permitiendo crear vistas para distintos tipos de dispositivos o plataformas; este patrón permite lograr mayor claridad de diseño, facilitando el mantenimiento, la escalabilidad del producto, y permitiendo la elaboración de las diferentes capas por diferentes equipos simultáneamente. La capa de vista es la responsable de la presentación del dominio de negocio, más concretamente, la que maneja la visualización de la información; el modelo es el encargado de toda la lógica y estado del dominio de negocio, en otras palabras, el control de los datos; el controlador por su parte se vincula al flujo de control, la navegabilidad y el estado de la entrada del usuario. De forma general este tipo de arquitectura se basa en que desde la vista, en el navegador Web del cliente, se hace alguna petición que es procesada por el controlador, y este se vincula al modelo para el uso de los datos.

La arquitectura de capas por otro lado define como estructurar el modelo de diseño en capas, que pueden estar distribuidas físicamente en distintos lugares, de forma tal que los componentes de una capa solo pueden hacer referencia a los componentes en capas inmediatamente inferiores

Por su parte la arquitectura basada en componente define el software como la unión de distintos componente, donde cada uno de estos es una unidad de composición con interfaz propia. Se les denomina unidad de composición y no de construcción debido a que no es preciso confeccionarlo, pueden ser comprados.

La arquitectura orientada a servicios (SOA, Service Oriented Architecture) se basa en la utilización de recursos que estén interconectados en una red, los cuales se conciben como servicios, accesibles por terceros a través de interfaces estándares. Existen en este modelo tres actores principales, el proveedor del servicio, el registro del servicio y el solicitante del servicio. Un componente podría considerarse en esta arquitectura un servicio, que puede ser publicado, descubierto e invocado de

Capítulo I. Fundamentación teórica.

forma dinámica. Este modelo de software brinda un bajo acoplamiento entre componentes y gran flexibilidad ante cambios

La arquitectura general del sistema Web para el evento de informática del año 2009 estará orientada al patrón MVC, por las ventajas que aporta de organización, reutilización del código y escalabilidad.

1.5 Diseño de software.

Existen criterios incorrectos acerca de que la arquitectura de software y el diseño de software son lo mismo, pero tal afirmación no es verídica puesto que la arquitectura se encuentra en un nivel de abstracción por encima del diseño, o es simplemente otro paso en el proceso de desarrollo de software, por otro lado el diseño se centra en aspectos más específicos relacionados con los subsistemas, las estructuras de clases, relaciones entre estas, así como los algoritmos a utilizar; al respecto Dewayne Perry, uno de los fundadores de la disciplina Ingeniería de Software, bosquejó la diferencia entre arquitectura y diseño como sigue, “La arquitectura concierne a un nivel de abstracción más elevado; 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 modos y los tipos. En cuanto a la composición, la de la arquitectura es de grano grueso, la del diseño es de fina composición de procedimientos; las interacciones entre componentes en arquitectura tienen que ver con un protocolo de alto nivel, mientras que las del diseño conciernen a la manera en que interactúan” (REYNOSO 2006).

1.5.1 Patrones de diseño.

Los patrones de diseño son básicamente métodos generales para resolver diferentes problemas relacionados con el diseño. Estos definen soluciones a problemas comunes. No fueron creados por personas que se propusieron escribir esquemas de diseño, sino que son soluciones que han sido implementadas una y otra vez por distintos equipos de desarrollo en diferentes proyectos. Estas soluciones reutilizables son denominadas patrones de diseño en las comunidades de desarrollo de software debido al libro “Design patterns: elements of reusable object-oriented software” (GAMMA *et al.* 1995) de Erich Gamma, Richard Helm, Ralph Johnson y John Vlissides (conocidos como el grupo de los cuatro), quienes después de años creando software se percataron de modelos de solución que se repetían para problemas de una misma índole, y basados en estos diseñaron alternativas para resolver problemas frecuentes de forma rápida y robusta, dando surgimiento a los patrones de diseño.

Christopher Alexander, quien creó el término “lenguaje de patrón”, define que: “Cada patrón describe un problema que ocurre una y otra vez en nuestro entorno, para describir después el núcleo de la

Capítulo I. Fundamentación teórica.

solución a ese problema, de tal manera que esa solución pueda ser usada más de un millón de veces sin hacerlo siquiera dos veces de la misma forma” (ALEXANDER 1979).

En el presente trabajo se utilizarán cinco patrones de diseño denominados patrones de asignación de responsabilidad (patrones GRASP, General Responsibility Assignment Software Patterns), ya que estos están basados en la recopilación del conocimiento de los expertos en desarrollo de software, para solucionar problemas comunes que se encuentran presentes en este proyecto. Estos patrones son, Experto, Creador, Alta Cohesión, Bajo Acoplamiento y Controlador.

El patrón Experto plantea que cada responsabilidad debe ser cumplida por la entidad que cuenta con la información necesaria para tal fin. Este será usado porque es una práctica muy buena para asistir a un diseño membrudo de las aplicaciones. El patrón Creador por su parte enuncia que se debe encargarse de crear una instancia de una clase solamente a aquella clase que agrega, contiene, registra o utiliza los objetos de la nueva clase, o que posee los datos de inicialización de la clase a crear, este modelo es un buen diseño de programación orientado a objetos (POO) que brinda una mayor claridad, encapsulamiento y reutilización. El esquema de Bajo Acoplamiento explica que los sistemas de este tipo son aquellos que sus partes no tienen dependencias fuertes con muchas otras partes, situación que podría provocar que los cambios en las instancias de las que se dependen afectarían a estas, dificultándose la reutilización de código. La Alta Cohesión se manifiesta en que las clases con responsabilidades estrechamente relacionadas no realicen un trabajo enorme, sino que se agrupen las clases según las responsabilidades afines, delegando las otras a otras clases, ya que sino sería muy difícil de comprender, dificultarían la reutilización, serían muy difíciles de conservar, así como les afectarían constantemente los cambios. El patrón Controlador designa la responsabilidad del manejo de los mensajes y eventos del sistema a una clase que represente al sistema global, un rol, o un caso de uso, no es encargado de realizar ninguna tarea del sistema, sino delegar y controlar estas, ganando así en más posibilidades de reutilización y organización del código.

Este conjunto de patrones que serán usados permitirán desarrollar un sistema con grandes potencialidades, ya que son considerados indispensables para realizar un buen diseño en los sistemas orientados a objetos. Serán empleados conjuntamente con los patrones utilizados por el CMS Drupal, los patrones: Singleton, Decorator, Observer, Bridge, Chain of Responsibility, Command (CORAI).

1.6 Herramientas de desarrollo.

Para el desarrollo del proyecto del que este estudio es clave, se hace necesario la utilización de herramientas que de una manera u otra contribuyen al buen desenvolvimiento del mismo, herramientas que aportan beneficios concretos y que serán especificados en cada caso.

1.6.1 Servidores de aplicaciones Web.

Para hacer disponibles los sistemas Web bajo el protocolo HTTP es necesario el uso de programas informáticos que sirvan como servidores de aplicaciones Web en dicho protocolo, software encargado de atender las peticiones de los navegadores utilizados por los clientes y enviar las respuestas correspondientes.

Existen distintas aplicaciones que cumplen la función de servidor de aplicaciones, entre las más utilizadas en la red de redes están Cherokee, Lighttpd, Internet Information Services (IIS) y Apache.

Cherokee es un servidor de aplicaciones Web publicado bajo licencia GNU GPL, escrito en el lenguaje C, proyecto que desarrolla una nueva implementación de este tipo de aplicaciones basado en características de las que Apache carece debido a su diseño original. Tiene un diseño mixto que combina las características de servidores basados en sockets con hilos, en busca de obtener beneficios de ambos modelos y minimizar los aspectos negativos. Para su implementación, se ha puesto especial interés en la velocidad, flexibilidad y capacidad de ser empotrado. Actualmente este producto no tiene documentación suficiente.

Lighttpd viene a ser una de las promesas entre los software servidores de aplicaciones del protocolo http, software que bajo licencia BSD a sido diseñado para profundizar en importantes aspectos como la seguridad, velocidad y flexibilidad, al cual renombradas revistas digitales (HAREM) sitúan en la cúspide de los servidores Web, por su gran efectividad utilizando los recursos de sistema, y factores como CGI, FastCGI, SCGI, Auth, Output-Compression, reescritura de URL y otras funcionalidades. La desventaja que tiene este producto es que no tiene tanta documentación como su rival apache.

Otro de los servidores de aplicaciones Web más utilizados es el Internet Information Services, software desarrollado por Microsoft que integra una serie de herramientas para la administración de sitios Web, y que además incluye otros servicios de Internet para la plataforma Windows. Comúnmente es conocido este software como Internet Information Server, debido al servicio incluido en el paquete para hacer función de servidor de aplicaciones. Este producto, como todos los de Microsoft, no está bajo licencia libre, por lo que es desechado entre las opciones de este estudio.

Apache HTTP Server 2.2 es un software libre y de código abierto, bajo licencia Apache, desarrollado como servidor de aplicaciones Web, al cual el Observatorio Tecnológico del Ministerio de Educación y Ciencia de España coloca en un 65% de uso entre los servidores de Internet (FERNANDEZ), el cual hizo su aparición desde Abril del 1995, con las principales metas de velocidad, simplicidad, multiplataforma y facilidad del desarrollo distribuido.

Capítulo I. Fundamentación teórica.

Aunque existe gran competitividad entre Lighttpd y Apache, se adoptará este último para el evento Informática 2009, por la madurez, experiencia en el desarrollo y amplia documentación, aunque se recomienda analizar el estado de Lighttpd para ser utilizado en posteriores versiones.

1.6.2 Entornos de desarrollo

Zend Studio Client 5.5 es una herramienta creada por la compañía Zend, de gran prestigio en los equipos de desarrollo de software. Es un entorno de desarrollo integrado (IDE) utilizado como editor de texto para programar usando el lenguaje PHP, tiene varias funcionalidades como lo son el completamiento de código, correr script paso a paso, permite depurar y corregir errores en tiempo de corrida, administración de proyectos y ficheros, el uso de plantillas de código, el realce de sintaxis con el uso de colores, formateo de código, permite ejecutar código script sin necesidad de utilizar un navegador Web, así como integración con programas controladores de versiones. Debido a que es desarrollado en el lenguaje Java funciona sobre todas las plataformas para la que este lenguaje tiene la maquina virtual. No es seleccionado para este proyecto puesto que tiene licencia privativa, pero se recomienda velar los cambios de licencias que puedan ocurrir.

PDT Project 1.1 es un IDE basado en la plataforma Eclipse, contiene todos los componentes de desarrollo necesario para trabajar con PHP. Es un producto con una interfaz muy intuitiva y fácil de aprender, licenciado bajo Eclipse Public License, la cual no es privativa. Tiene un editor sensible al contexto, con resaltamiento de código, auto completamiento y explorador de ficheros.

Dev-PHP es uno de los destacados IDE para el lenguaje PHP, que bajo licencia GNU GPL es usado por comunidades de desarrollo por lo potente y rápido que es, al igual que por los diversos beneficios que brinda, como: navegador de clases, visor de script, integración con analizadores gramaticales, depuración de código, entre otras. El aspecto que lo deja fuera de la elección, es que solo funciona sobre plataformas de Microsoft Windows.

Bluefish 1.0.7 es un potente editor de texto de licencia GNU GPL, muy utilizado por las comunidades de software libre para programar en gran variedad de lenguajes de programación, entre ellos el utilizado por este proyecto, PHP. Este IDE está elaborado especialmente para el desarrollo Web. Existen versiones del mismo para los sistemas operativos más utilizados.

Para la etapa de desarrollo del proyecto, se propone utilizar Bluefish 1.0.7, puesto que aunque existe gran variedad de herramientas con muchas prestaciones como IDE para el lenguaje PHP, Bluefish se encuentra entre los más competitivos de los pertenecientes a licencias libres.

Capítulo I. Fundamentación teórica.

1.6.3 Software cliente del Sistema Gestor de Base de Datos.

Todo desarrollo de software que involucre la utilización de un Sistema Gestor de Base de Datos (en lo adelante SGBD) requiere entre los equipos de desarrollo la utilización de un cliente para dichos sistemas. Entre los sistemas de este tipo más utilizados se encuentran:

MySQL Navigator es un software de SGBD, que tiene licencia GNU GPL, utilizado para manipular las bases de datos de servidores MySql sobre los sistemas operativos más populares, desarrollado desde el año 2001, con una producción estable.

MySQL-Admin está desarrollado como ambiente Web en PHP, característica que lo hace multiplataforma, con licencia GNU GPL y con funcionalidades de un cliente de bases de datos profesional. Desarrollado desde el año 2006.

PhpMyAdmin 2.11.1 es una herramienta bajo licencia GNU GPL escrita en PHP que permite administrar el SGBD MySql con la simple utilización de un navegador Web. Permite operar el SGBD mencionado desde cualquier lugar donde se tenga acceso vía HTTP a la maquina que lo hospeda, así como todas las funcionalidades propias de una aplicación cliente de base de datos profesional, modificar la estructura interna de las bases de datos, ejecutar consultas, realizar salvos o restaurarlas, entre otras.

Este estudio propone utilizar en la etapa de desarrollo el PhpMyAdmin o el MySQL-Admin, ya que ambas son potentes herramientas con prestaciones muy similares, aunque se recomienda el PhpMyAdmin pues es un producto más probado.

1.6.4 Herramientas Case

Las herramientas CASE son programas informáticos muy necesarios para los equipos de desarrollo de software, para brindar soporte de alto nivel a las actividades de modelado y todo lo que de estas puede derivarse o automatizar tareas repetitivas que suelen ser tediosas y que no requieren de intelecto para su ejecución, puesto que son resultado de una función de transformación.

Visual Paradigm for UML 6.0 es una herramienta CASE que permite mediante el uso del UML organizar y controlar el desarrollo de software, especialmente aquellos que por su complejidad deben estar bien documentados y organizados. El uso de esta herramienta ayuda a los diseñadores, programadores, probadores, planificadores, y administradores de proyecto a tener una visión común de las etapas de desarrollo de un proyecto. Esta herramienta ayuda a establecer disciplinas de trabajo en equipo, a rastrear los progresos de los mismos, mediante gráficos, así como a gestionar los planes de negocio. Permite realizar tanto ingeniería directa como inversa. Es una herramienta colaborativa ya que permite trabajar sobre el mismo proyecto múltiples usuarios concurrentemente.

Capítulo I. Fundamentación teórica.

Existe una herramienta CASE llamada Rational Rose que está catalogada como una de las mejores, pero no se tomará en cuenta como propuesta de este trabajo porque es un software con licencia privativa, y solo funciona sobre sistemas operativos Microsoft Windows. Tampoco se tendrán en consideración otras herramientas CASE, debido a que aunque existe un gran número de estas, ninguna es tan profesional, hasta el momento en que se realiza este estudio, como el Visual Paradigm for UML 6.0, razón por la cual resulta la herramienta que propone utilizar.

1.6.5 Controlador de versiones.

El desarrollo de un producto transcurre por diferentes etapas, períodos en los que se obtienen versiones del mismo. El proceso de gestionar los diversos cambios que se realizan sobre los elementos del producto es denominado control de versiones. Los sistemas para el control de versiones permiten la administración de las distintas versiones del producto. Los sistemas de control de versiones aportan a los equipos de desarrollo organización y control, les facilita el trabajo colaborativo, ya que los integrantes del equipo pueden trabajar sobre el mismo fichero, o actualizar la copia en que trabajan por los cambios hechos por otro miembro. Existen diversos software para este propósito, los más utilizados, y por analogía los más profesionales, serán mencionados y caracterizados a continuación.

Microsoft Visual SourceSafe es una herramienta de control de versiones similar a todas las de este tipo, el aspecto más significativo y por el cual queda excluido de este estudio es que tiene licencia privativa y solo funciona sobre sistemas operativos Microsoft Windows.

CVS (por sus siglas en inglés, Concurrent Versions System) y Subversión son herramientas muy similares, ambas son software libre, funcionan sobre todos los sistemas operativos más difundidos y realizan básicamente lo mismo, con la diferencia que en Subversión los archivos versionados no tienen cada uno un número de revisión independiente, sino que todo el repositorio tiene un único número de versión que identifica un estado de todos los archivos del repositorio en cierto momento.

El controlador de versiones utilizado para esta investigación y el propuesto para la etapa de desarrollo es el Subversión 1.1, basado en que este realiza las mismas funciones que el CVS, pero hace un mejor tratamiento del renombrado de ficheros.

1.6.6 Navegadores de Internet.

Entre los navegadores más utilizados por los ínter nautas se encuentran el Microsoft Internet Explorer 6.0 y el Mozilla Firefox 2.0.0.11.

El navegador de Microsoft resultada uno de los productos de su índole más presentes en Internet producto que viene junto a los sistemas operativos Microsoft Windows, pero es considerado uno de los

Capítulo I. Fundamentación teórica.

navegadores que menos respeta los estándares, razón por la cual los equipos de desarrolladores Web tienen que dedicar etapas del desarrollo a adecuar las interfaces de los sistemas a este navegador.

Mozilla Firefox 2.0.0.11, o sencillamente Firefox, como es conocido por todos los equipos de desarrollo de software, es un navegador Web creado por la Fundación Mozilla, bajo la licencia Mozilla Public License. Está categorizado por las comunidades de software como la aplicación más útil de su tipo, debido a que permite integración con muchísimas extensiones (plug-in) para desarrolladores y otros tipos de intereses, que aportan funcionalidades como son la validación de código javascript, o HTML. Un estudio estadístico realizado por la W3Schools(W3SCHOOLS) en el año 2007 determinó que en ese año el uso de Firefox ascendió de un 31,0% a un 36,3%, afianzándose como el navegador más utilizado en Internet, en Enero del año 2008 esta cifra subió a 37,2%.

Los datos anteriormente expuestos, junto a que Mozilla Firefox es uno software de su tipo que más cumple los estándares definidos por la W3C, son las causas por la que se recomienda que el mismo sea el navegador Web a utilizar en los períodos de desarrollo del proyecto.

1.7 Lenguajes.

En este estudio se diseña un sistema en base a algunos lenguajes utilizados por los equipos de desarrollo de software, mediante los cuales se irá elaborando el producto final deseado, a los cuales se hace alusión a continuación.

1.7.1 Lenguaje de programación PHP.

PHP es un lenguaje de programación interpretado, usado normalmente para la creación de páginas Web dinámicas. Es un acrónimo recursivo de "PHP Hypertext Pre-processor". Este lenguaje tiene una sintaxis muy similar a lenguajes de programación comunes como C y Perl, lo que facilita crear aplicaciones complejas con una curva de aprendizaje relativamente corta. Es un lenguaje multiplataforma, es decir, sus funciones no depende del sistema operativo en que se ejecuta el script, excepto algunas funcionalidades nativas, de esto se encarga el interprete de PHP incluido en los servidores de aplicaciones, que si son específicos para cada sistema operativo. Este lenguaje de programación puede interactuar con los servidores de aplicaciones Web más populares ya que existe en versión CGI, módulo para Apache, e ISAPI.

Actualmente este lenguaje se encuentra en la versión 5, entre sus características se encuentra que es un lenguaje libre, permite técnicas de programación orientado a objetos, multiplataforma, débilmente tipificado ya que no requiere la definición de tipos de variables, capacidad de conexión con la mayoría

Capítulo I. Fundamentación teórica.

de los SGBD, aunque destaca su conectividad con MySQL, permite tratar las excepciones; se encuentra en desarrollo la versión 6 del mismo.

Entre las desventajas de este lenguaje podemos encontrar que no posee una abstracción de base de datos estándar, sino bibliotecas especializadas para cada SGBD; no posee adecuado manejo de internacionalización, debido a que es un lenguaje interpretado no es posible la optimización de respuestas en los servidores y por estas mismas características de ser script algunos desarrolladores que abogan por la separación del código y las interfaces de usuario lo rechazan, aunque este problema puede ser resuelto con buenas prácticas de programación.

El modo de funcionamiento de los servidores de aplicaciones que soportan este lenguaje de script es el siguiente: cuando un cliente hace una petición al servidor este procesa el script y genera el contenido de la página que se envía como respuesta al cliente.

Con este lenguaje se han desarrollado un grupo importante de las aplicaciones más utilizadas en Internet, ejemplos de estas son:

- Drupal
- Joomla
- MediaWiki
- Moodle
- PHPMyAdmin
- PHP-Nuke
- PHPPgAdmin
- PostNuke
- Smarty
- Wikipedia
- WordPress

La TIOBE Company, una compañía que entre otras labores publica en su sitio Web los lenguajes de programación más populares de cada mes, este lenguaje script es actualmente uno de los lenguajes de programación más preferidos por los desarrolladores, en el año 2007 ocupó el quinto puesto, y en lo que va de año 2008 ha subido un peldaño (COMPANY, TIOBE), dato que es reafirmado por la herramienta Google Trend, así como por Tim O'Reilly, fundador y presidente de la famosa editorial O'Reilly en el sitio Web de la empresa (O'REILLY), partiendo del análisis de los libros de programación que más vende su compañía.

El lenguaje PHP ha sido seleccionado para ser el lenguaje en el que se basará el diseño del sistema debido a que es el lenguaje con el cual está desarrollado el CMS seleccionado por los clientes, al cual será necesario agregarle determinados módulos. Los datos anteriores sirvieron para determinar la factibilidad del uso de este lenguaje de programación.

1.7.2 Lenguaje de modelado unificado.

El Lenguaje de Modelado Unificado (UML, Unified Modeling Language) es un lenguaje para visualizar, construir, especificar y documentar los artefactos de un sistema software. Es un estándar de la Object Management Group (OMG).

Capítulo I. Fundamentación teórica.

La utilización de este lenguaje de modelado facilita la gestión del desarrollo de software, permitiendo ocultar o exponer detalles según sea necesario, elaborando diagramas de abstracciones de sistemas o diseñando procesos llegando a especificidades de cada lenguaje de programación. Ayuda también a mantener la consistencia entre los artefactos del sistema: requisitos, diseños e implementaciones. Como resumen se puede afirmar que el modelado visual ayuda a mejorar la capacidad del equipo para gestionar la complejidad del software, permitiendo a los equipos de desarrollo lograr mejor entendimiento de los procesos una vez graficados.

UML 2.0, que será el utilizado en este estudio, cuenta con trece tipos diferentes de diagramas (AMBLER 2005). Para lograr una mejor comprensión, es útil dividirlos en tres categorías:

Los diagramas de estructura describen los elementos que deben existir en el sistema:

- Diagrama de clases
- Diagrama de componentes
- Diagrama de despliegue
- Diagrama de estructura compuesta (a partir de UML 2.0)
- Diagrama de objetos
- Diagrama de paquetes

Los diagramas de comportamiento se utilizan para modelar los procesos:

- Diagrama de actividades
- Diagrama de casos de uso
- Diagrama de estados

Los diagramas de interacción son un subtipo de diagramas de comportamiento, que enfatizan sobre el flujo de control y de datos:

- Diagrama de colaboración
- Diagrama de tiempos (a partir de UML 2.0)
- Diagrama de secuencia
- Diagrama de vista de interacción (UML 2.0)

1.8 Sistemas manejadores de contenidos.

Los sistemas de gestión de contenidos (Content Management Systems o CMS) son software que se utilizan principalmente para facilitar la gestión de contenidos en la Web, ya sea en Internet o en una intranet, es por eso que son conocidos también como gestores de contenido Web (Web Content Management o WCM), han alcanzado gran popularidad, como lo son las Wikis, y los sistemas de trabajo colaborativo (groupware).

La funcionalidad de los sistemas de gestión de contenidos se pueden dividir en cuatro categorías fundamentalmente: creación de contenido, gestión de contenido, publicación y presentación, estos definidos bajo flujos de trabajos que varían según el CMS en cuestión.

Estos sistemas se desarrollan de forma genérica, para que puedan ser adaptados a las condiciones de cada interesado en darle uso, permitiendo la gestión de aspectos como son los menús de navegación o la jerarquía de las páginas, añadiendo enlaces de forma automática. También permiten adicionar módulos a ellos, para garantizar la escalabilidad y adaptabilidad al entorno donde se le de uso. Todo esto bajo patrones que los diseñadores hayan definido.

1.8.1 ¿Qué CMS elegir?

Existe tanta variedad de CMS como de lenguajes humanos prácticamente, por lo que este estudio se ha dado la tarea de indagar sobre los más utilizados.

Cuando las personas se proponen comparar determinados objetos, siempre se corre el riesgo de dejarse llevar por la subjetividad y emitir criterios personalizados según las experiencias de cada cual, es por esto que la comparación siguiente se basará estrictamente en datos recolectados por organizaciones dedicadas al desarrollo de los CMS, como Plain Black Corporation (CORPORATION) y OpenSourceCMS(COLLECTIVE), y de datos extraídos de los propios sitios Web de cada producto. Los datos en los que se basa este análisis se muestran en la tabla del [anexo1](#).

Los expertos del jurado para el concurso anual desarrollado por la empresa Pack Publishing sobre el desarrollo de los CMS, determinaron al CMS Drupal como el ganador del premio CMS Open Source del año 2007 (LTD) otorgado por esa compañía, quedando por encima de otros competitivos CMS, como los que se comparan en el anexo anteriormente citado, ya que se caracterizó según opinión de los usuarios como “uno de los desarrollos más activos y mejores organizados en estos dos o tres años” (CORAI).

Al analizar todos los CMS comparados en la tabla del anexo se concluye que para el momento de este estudio cualquiera de los productos analizados puede ser utilizado, ya que todos cumplen los requisitos básico necesarios, pero el más recomendable a usar lo constituye el CMS Open Source

Capítulo I. Fundamentación teórica.

Drupal, por tener un desarrollo más estable, una comunidad de desarrolladores considerablemente amplia, la existencia de la documentación necesaria, de listas de distribución organizadas y asistidas por los desarrolladores fundamentales de este producto, además de que es un producto más notorio que el resto de los CMS Open Source, un ejemplo del buen renombre que tiene este CMS lo puede ser que sea el elegido para las intranet de empresas como la transnacional IBM (WEITZMAN *et al.*) , el proyecto Ubuntu (UBUNTU), la Fundación Mozilla (FOUNDATION), y otros importantes sitios empresariales como el de Yahoo! Research (RESEARCH) y la corporación AOL (COMPANY, AOL) además de sitios de gobiernos como el de la NASA (NASA), entre otros (BENC).

1.9 Los sistemas gestores de bases de datos.

Los sistemas gestores de base de datos (SGBD) son un tipo de programa informático muy específico, utilizado con la finalidad de servir de interfaz entre las bases de datos, los usuarios y las aplicaciones que utilicen dichos datos. Este tipo de software es utilizado mediante tres tipos de lenguajes, el lenguaje de definición de datos (DDL, Data Definition Language), el lenguaje de manipulación de datos (DML, Data Manipulation Language) y el lenguaje estructurado de consultas (SQL, Structured Query Language).

El propósito de este tipo de aplicaciones es servir de intermediario entre las bases de datos y los entes, ya sean usuarios o programas, que hagan uso de las mismas, permitiendo un grado de abstracción tal que no se necesite saber detalles del almacenamiento físico de los datos para hacer uso de ellos, lograr independencia entre los datos y las aplicaciones, la no existencia de redundancia en la información, consistencia entre los datos que tengan relación, existencia de permisos para garantizar la seguridad de los datos, integridad de la información almacenada, sistemas de respaldo y recuperación, control de concurrencia, y minimizar los tiempos de acceso a los datos.

- Ventajas:
 - Fácil de manejo de grandes volúmenes de información.
 - Rápido acceso a los datos.
 - Seguridad de la información.
 - Eliminación de redundancia en la información.
- Inconvenientes:
 - Constituye un costo más en el proyecto la adquisición de hardware y software necesario, así como el salario para las personas encargadas de brindar mantenimiento a base de datos.
 - Un mal diseño o problemas de seguridad puede provocar grandes pérdidas económicas.

Capítulo I. Fundamentación teórica.

Entre los SGBD más utilizados, estos datos se puede corroborar en Google Trends y en las comunidades de desarrollo, se encuentran el Oracle, MySQL, PostgreSQL y Microsoft SQL Server, este último y Oracle quedan excluidos de este estudio por utilizar licencias privativas.

PostgreSQL 8.0 (GROUP) se propone ser un sistema gestor de bases de datos de alto nivel, al margen de Oracle, Sybase IQ o Interbase, diseñado especialmente para base de datos de tamaño considerablemente grande, publicado bajo licencia BSD, que tiene las ventajas de soportar transacciones desde la versión 7.0, claves ajenas, disparadores, vistas y procedimientos almacenados, por el contrario de las versiones más antiguas de MySQL. Entre sus inconvenientes está que consume bastantes recursos de sistema, según opinión de muchos equipos de desarrollo que lo han usado, lo han caracterizado 2 o 3 veces más lento que MySQL y plantean que existen menos funciones en PHP para trabajar que en el caso PHP-MySQL.

MySQL 5.0 es un SGBD que desde la versión 3.23.19 está bajo licencia GNU GPL, el cual fue desarrollado básicamente con la intención de lograr mayores velocidades en el acceso a los datos, por lo que sacrificaron otros aspectos que influyesen en esta. Es un sistema que consume pocos recursos de la maquina en que se ejecuta. Tiene como ventajas que permite obtener un mayor rendimiento, ya que se logran mayores velocidades tanto al conectar con el servidor como al realizar consultas, especialmente las de selección de datos. Existen para este SGBD una amplia variedad de utilidades de administración (salvas de seguridad, recuperación de errores y otros). Con este SGBD se logra una mayor integración con PHP. No es aconsejable usar este SGBD para sistemas donde las operaciones en la base de datos se realicen sobre muchos datos y a través de consultas con amplio grado de complejidad.

En general, MySQL es más aconsejable para sistemas en que la velocidad es algo primordial, y la seguridad no sea muy importante (donde sea suficiente realizar salvas periódicamente para restaurar la base de datos tras una caída del servidor). En cambio, para sistemas más sensibles en los que la consistencia de la base de datos sea fundamental (bases de datos con información realmente elemental como bancos, registros penales, y otros.) PostgreSQL es una mejor opción pese a su mayor lentitud (AGUILAR and SUAU).

Debido a que en la aplicación Web a desarrollar será baja la concurrencia de modificación de datos y en cambio lo que respecta a lectura de datos será intensivo, MySQL resulta ideal en este caso, razones por la cual será el SGBD adoptado en este trabajo.

En este capítulo se realizó un estudio del estado del arte del sistema, en el cual no hubo resultados obtenidos, debido a que los sistemas existentes con este fin requieren de pagos por inscribirse y

Capítulo I. Fundamentación teórica.

disfrutar de sus beneficios, tampoco brindan información sobre los sistemas empleados, por lo que se decidió regirse por los resultados de la investigación sobre tendencias y comparaciones entre las distintas metodologías, tecnologías, técnicas, lenguajes y herramientas que son utilizados para elaborar software con las características deseadas.

Capítulo II. Caracterización del sistema.

En este capítulo se abordará lo referente al negocio a estudiar mostrando en detalles los procesos que serán objeto de automatización del mismo.

En lo adelante se modelarán diagramas basados en la metodología RUP, que parte de un estudio del negocio, permitiendo describir las actividades que se realizan en la actualidad sin la existencia del sistema.

Se plantea la propuesta del sistema, describiéndose los procesos de negocio actuales y la propuesta del sistema para la problemática descrita, haciéndose referencia al flujo de trabajo modelación del negocio, donde se definen y describen los actores, trabajadores y casos de usos del negocio y del sistema, dándose a conocer las características del sistema, con la captura de los requisitos funcionales, y las cualidades del mismo con los requerimientos no funcionales.

Se muestran los diagramas de casos de usos del negocio y del sistema, diagramas de actividades y el modelo de objetos general.

2.1 Objeto de estudio.

2.1.1 Objeto de automatización.

Será objeto de automatización todo el proceso informativo de la feria y la convención, el proceso de solicitud de los stand expositivos, donde los interesados y organizadores puedan ponerse de acuerdo sobre las condiciones, el envío de trabajos por parte de los interesados en participar en el evento a los organizadores del mismo, que en la actualidad se realiza de forma manual, donde los organizadores los ordenan por comisiones y los envían al comité científico al que pertenezca, actividad que resulta muy engorrosa en la actualidad debido al gran volumen de trabajos. En el diseño del sistema propuesto es automatizado este proceso, ya que se permitirá a los usuarios registrados en el sistema, el envío de trabajos directamente al comité científico al que corresponde el trabajo.

Otro de los procesos de automatización lo constituyen las solicitudes de participación como delegado en la convención, ya sea como espectador o expositor.

2.1.2 Información que se maneja.

En el desarrollo de la Convención y Feria Internacional de Informática que sesionará en el año 2009 se hace necesario poseer para un buen desenvolvimiento del evento, ciertas informaciones personales de los participantes de carácter obligatorio, como el nombre y apellidos, país al que pertenece, dirección

Capítulo II. Caracterización del sistema.

postal, correo electrónico, y otros datos opcionales como la ocupación, la entidad que representa, la URL personal, y el currículum personal, informaciones que se mantendrán de manera confidencial, a la que solo podrán acceder los organizadores del evento por el propio trámite organizativo.

Los trabajos presentados serán publicados y podrán ser accedidos por todos los usuarios del sistema, bajo los términos legales que se publiquen en el mismo.

Se controlará en el sistema a diseñar, información relativa a los intereses de cada usuario, para en todo momento mantener informado al mismo, sobre el estado de las actividades de su interés.

2.1.3 Propuesta de sistema.

La propuesta de este trabajo consiste en el análisis y diseño que permitirá desarrollar una aplicación Web, mediante la cual se logre una mayor participación en la Convención y Feria Internacional de Informática 2009, por usuarios virtuales que tengan interés en participar en este importante acontecimiento científico. En dicho sistema se proveerá a los organizadores de la herramienta necesaria para difundir la información, así como gestionar los trabajos presentados. Por otra parte las personas interesadas en participar como espectador o delegado en la convención podrán inscribirse al sistema libremente y a través de un pago electrónico, adquirir el derecho a la participación oficial en el evento.

Los usuarios de esta aplicación Web una vez autenticados en la misma podrán disfrutar de ciertas bondades como son acceder virtualmente a los salones de eventos para ver en estos las ponencias virtuales, las ponencias presenciales, las relatorías del evento y los paseos virtuales por la feria, entre otros beneficios.

2.1.4 Modelo de negocio.

La metodología RUP plantea como uno de los principales objetivos de la fase de inicio la definición de viabilidad, para determinar si se debe continuar con el desarrollo del software, tarea por la cual se hace necesario realizar un modelo del negocio o un modelo de dominio.

En este caso los procesos están bien definidos, razón por la cual se define un modelo de negocio, a su vez este sirve de ayuda en el momento de realizar el levantamiento de los requisitos que el cliente desea, permitiendo lograr una mejor comprensión sobre el entorno en que se tiene que desplegar la aplicación.

El modelo de negocios tiene como objetivo esencial permitir la comprensión total de la estructura y dinámica de la organización en la cual se va a implantar la aplicación informática, así como

Capítulo II. Caracterización del sistema.

comprender los problemas actuales de los clientes e identificar las mejoras, logrando que los usuarios finales y el equipo de desarrollo tengan una visión común sobre el problema a resolver.

2.1.5 Breve descripción del negocio.

La Convención y Feria Internacional de Informática es un evento internacional que se desarrolla en Cuba con una periodicidad de dos años, en el que se presentan diversas personalidades de renombre internacional, y toda una comunidad Iberoamericana atraída por temas científicos vinculados a la informática, telecomunicaciones, electrónica y automática.

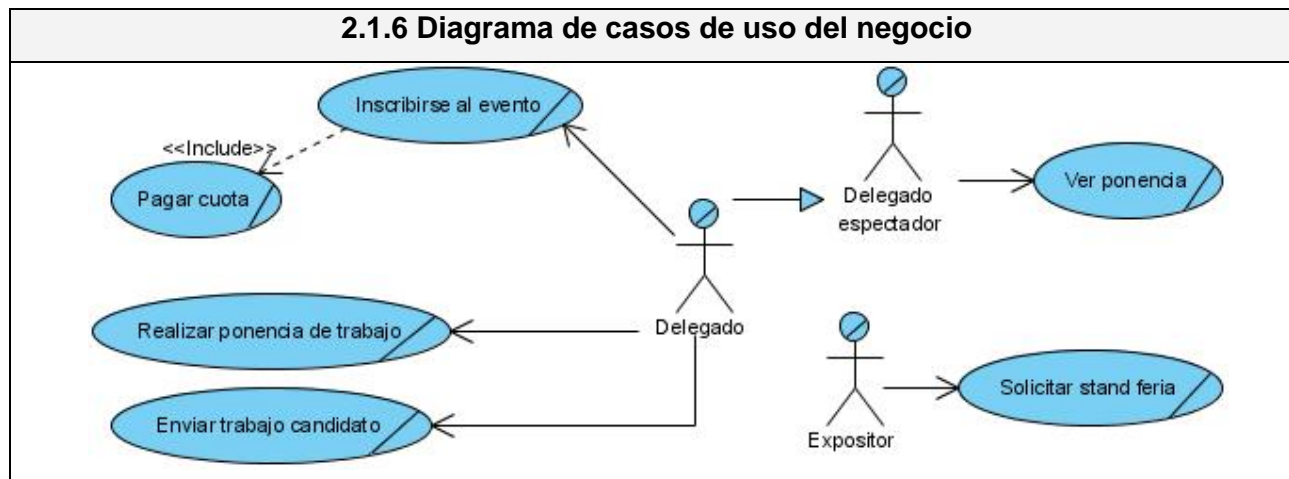
En cada ocasión se divide el evento en dos secciones que se realizan en ciertos momentos simultáneamente, la Feria, donde se exponen los temas más innovadores de las ramas presentes por empresas nacionales e internacionales, y se utiliza este momento para establecer importantes encuentros entre diversos empresarios del revolucionario mundo de las áreas mencionadas. La otra parte de este importante acontecimiento es la Convención, donde se presentan investigaciones de todas las personas interesadas que antes del comienzo de la misma enviaron su solicitud a los organizadores del evento y estos determinaron si constituían investigaciones con el grado de aporte suficientemente relevante para ser presentadas en este suceso científico.

En la convención se presentan los trabajos elegidos, mediante ponencias, actividades que al culminar permiten a los espectadores, con dudas o interés de profundizar en temas, platicar con los autores, propiciando el intercambio científico.

Actores del negocio	Justificación
Expositor	Asiste a la feria para presentar un trabajo o investigación en uno o varios stand, según la cuota que pague, donde interactúa con los visitantes y empresarios de las diferentes ramas presentes.
Delegado	Participa en la convención para presentar algún trabajo en una ponencia, paga una cuota por el derecho a ser delegado, lo que le permite la participación en las distintas actividades que se realizan en la convención y recibe un certificado como constancia de dicha participación.
Delegado espectador	Delegado que actúa como espectador mientras otros delegados presentan su trabajo, concurre a las ponencias de trabajos como espectador de los trabajos que se presentan, y como tal interactúa, si así lo desea, con los otros delegados sobre dudas o interés de un tema tratado.

Capítulo II. Caracterización del sistema.

Trabajadores del negocio	Justificación
Organizador de la feria	Encargado de organizar la feria y dirigir las comisiones de trabajo. Gestiona las solicitudes de participación en la feria. Reenvía las solicitudes de ponencias para la convención al comité científico correspondiente.
Comité científico	Rol encargado de evaluar los trabajos de una comisión específica, descartando antes del evento aquellos que no poseen la calidad requerida. Una vez empezada la convención su función es organizar las conferencias, las ponencias de trabajos, y las relatorías.



Al comprender los procesos de negocio que tienen lugar en la Convención y Feria Internacional de Informática, se determinó un grupo de casos de uso, como se aprecia en la figura anterior, que serán descritos a continuación:

Caso de uso “Inscribirse al evento”

Comienza cuando un posible participante se interesa por acudir a tan importante acontecimiento informático, busca información sobre el mismo para encontrar los datos necesarios para inscribirse y participar como delegado para presentar algún trabajo o no, y envía estos datos por correo electrónico o fax a los organizadores del evento, estos los verifican y confirman al interesado la aceptación o no, en caso positivo, se le notifica que debe proceder a pagar una de las diferentes cuotas de delegado.

Caso de uso “Enviar trabajo candidato”

Capítulo II. Caracterización del sistema.

Este caso de uso se inicia cuando una persona desea presentar algún trabajo en la convención, situación por la que envía el trabajo a los organizadores del evento, y estos a su vez la envían al comité científico correspondiente, que revisa el trabajo y determina si el mismo tiene la calidad suficiente como para ser presentado. Al analizar el trabajo informan al autor la decisión y en caso de ser aprobado lo inscriben en el evento si no está ya inscrito.

Caso de uso “Realizar ponencia de trabajo”

Comienza cuando a cada delegado le toca su turno de realizar la ponencia de su investigación, y al finalizar esta el comité organizador realiza preguntas y posteriormente le brinda esta posibilidad al resto de los delegados que están como espectadores.

Caso de uso “Ver ponencia”

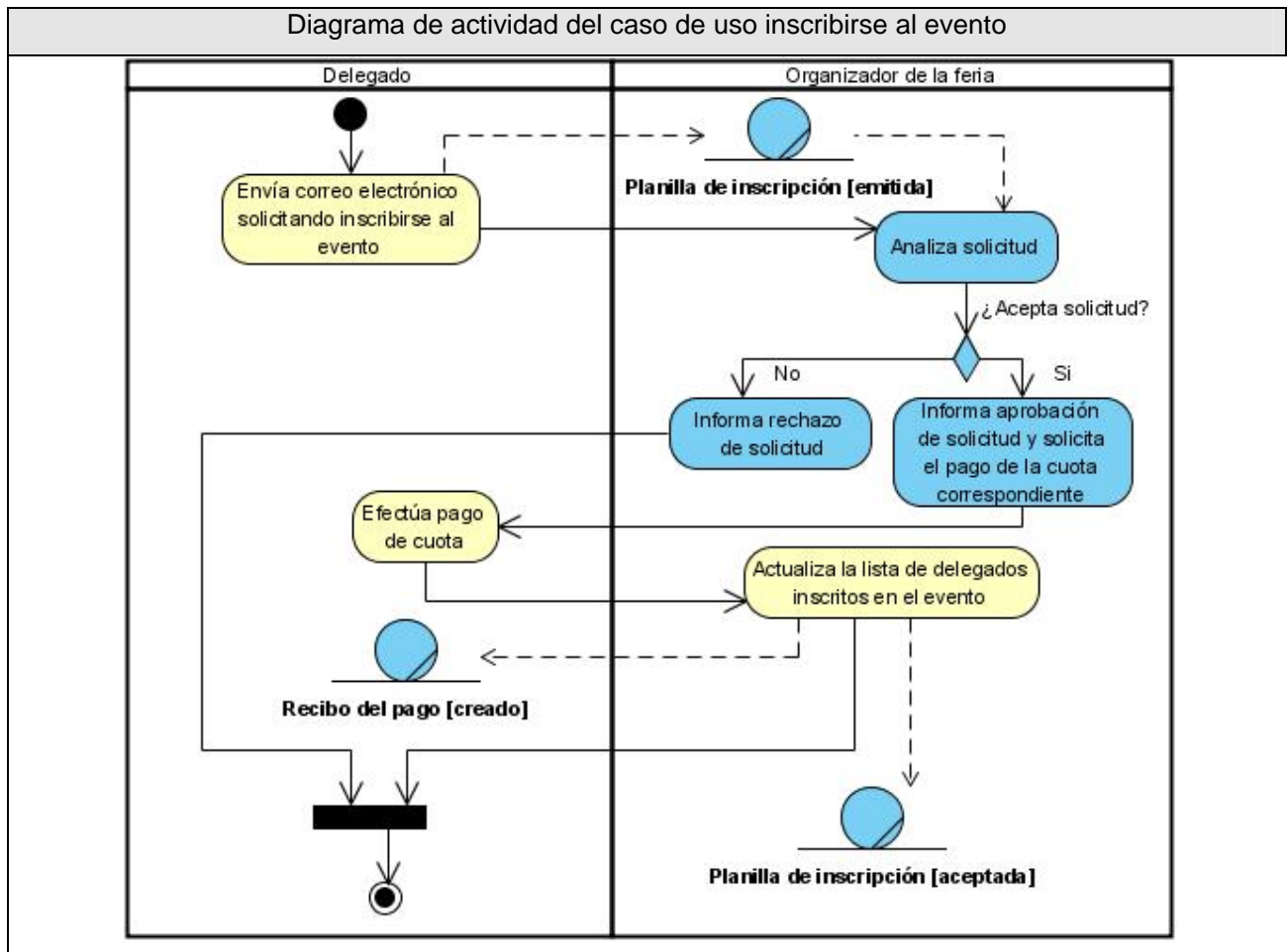
Este caso de uso comienza cuando algún delegado espectador se interesa por conocer las peculiaridades de un trabajo en particular o los trabajos de una comisión determinada de la convención, y asiste a la ponencia de estos, donde al finalizar puede intercambiar con cada autor.

Caso de uso “Solicitar stand expositivo”

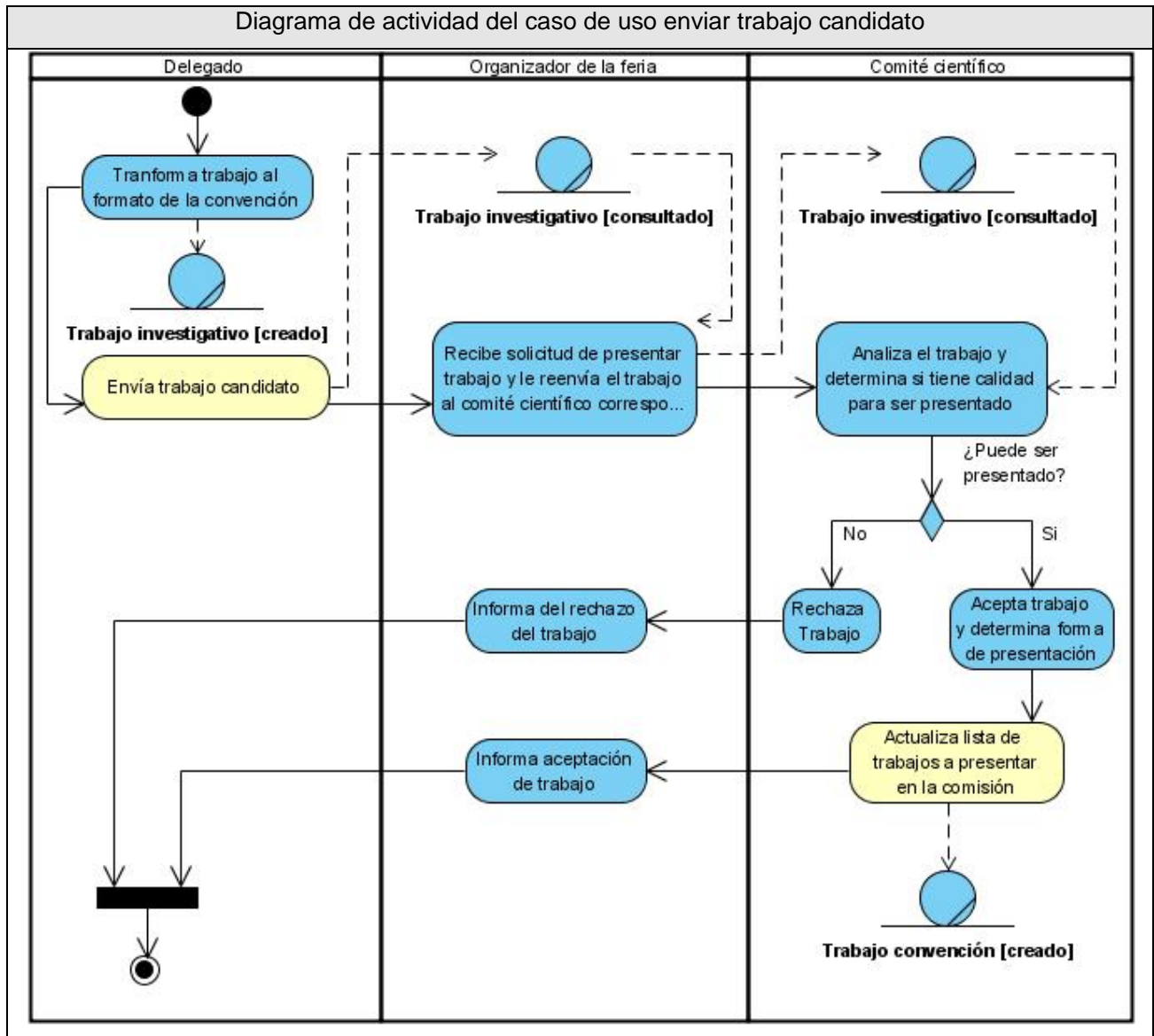
Este caso de uso es iniciado por los expositores de la feria, que solicitan un área del recinto ferial para colocar el stand de la empresa o institución que representen. Estos expositores pagan una cuota que está en dependencia de sus requerimientos del stand, y al pagar dicha cuota adquieren el derecho de exponer en los días que sesiona la feria sus productos, donde se exponen diversos temas de gran relevancia en las ramas informática, telecomunicaciones, electrónica y automática, y se realizan importantes encuentros comerciales entre los distintos participantes.

Cada uno de estos casos de uso queda gráficamente detallado en los siguientes diagramas de actividades.

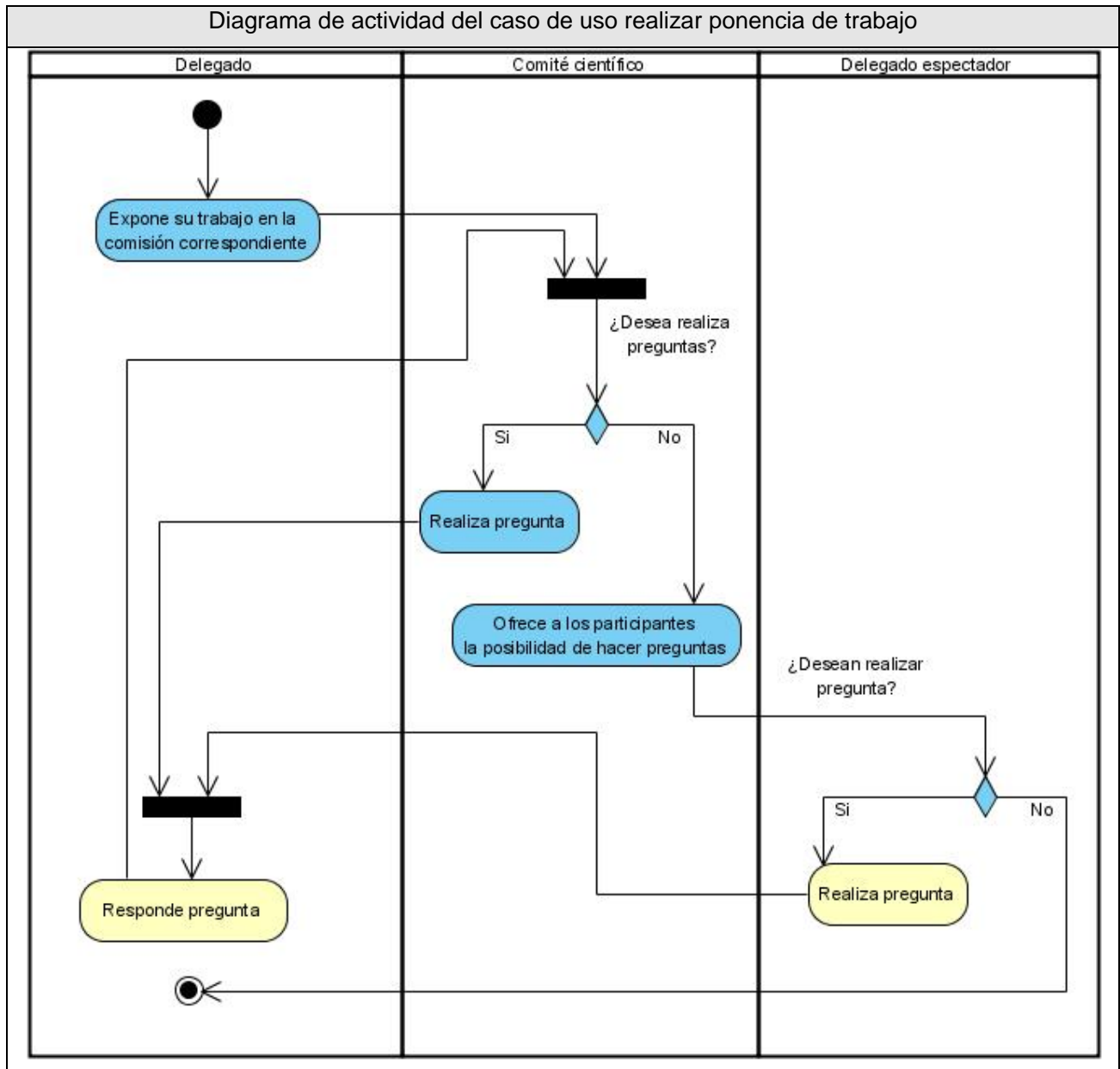
2.1.7 Diagramas de actividades.



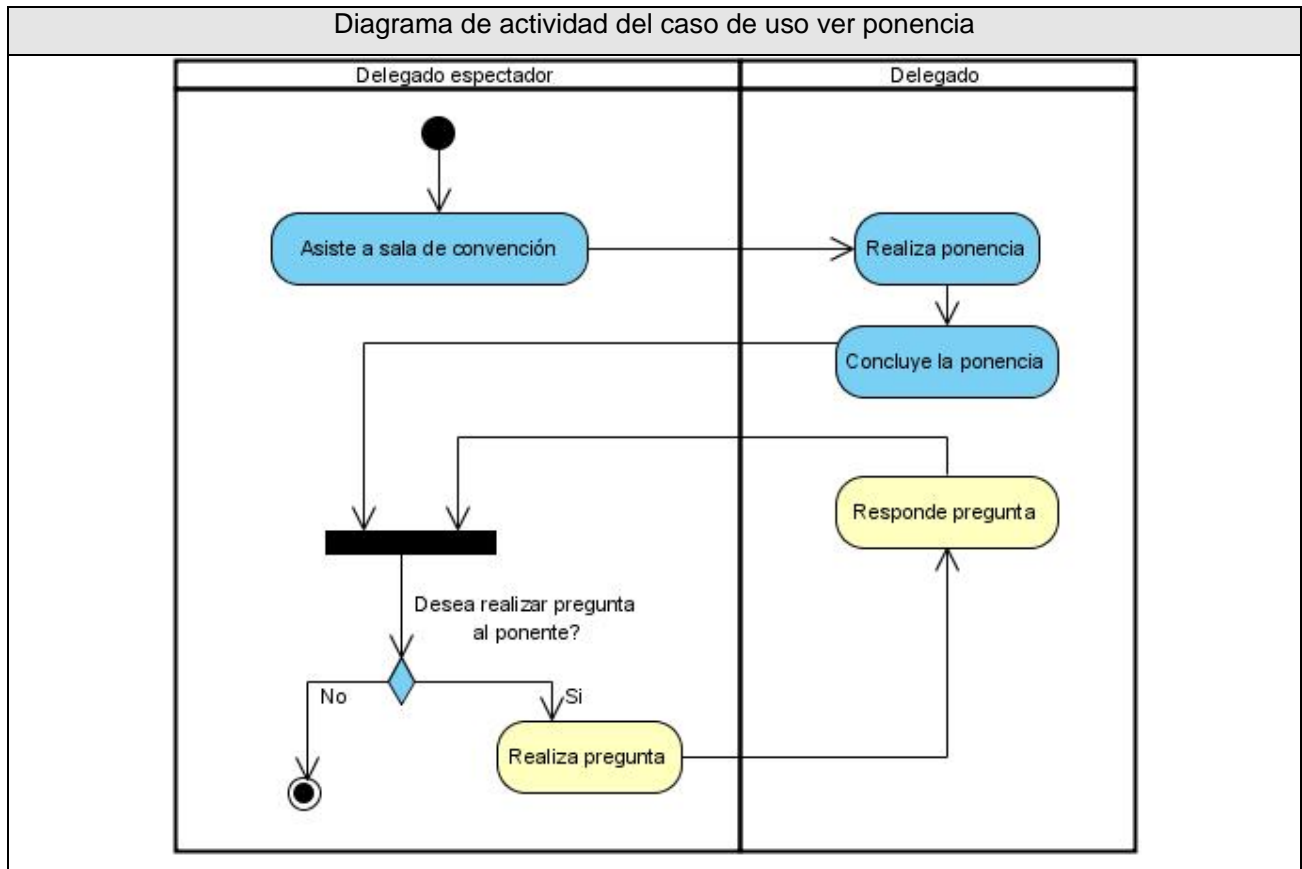
Capítulo II. Caracterización del sistema.



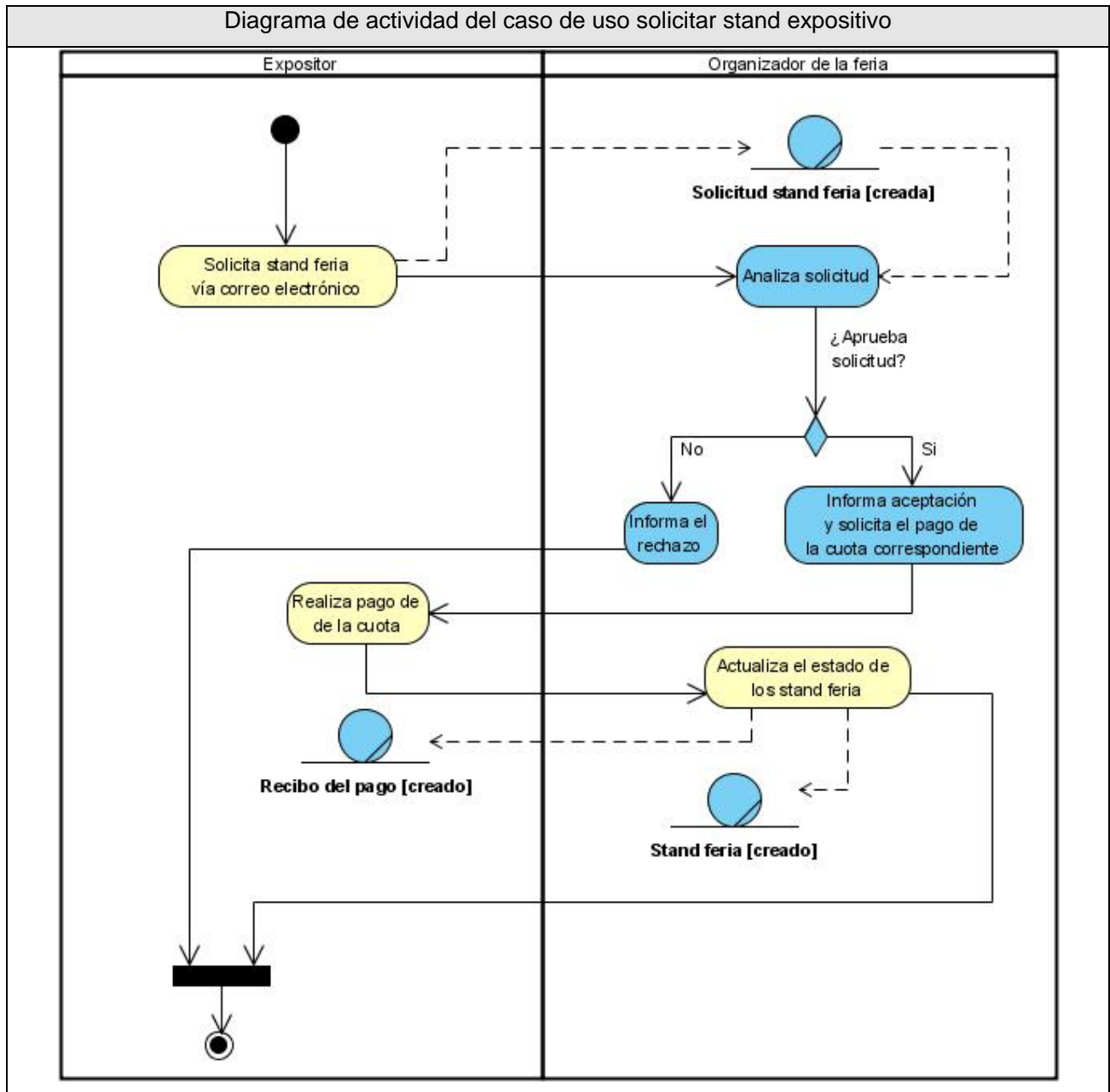
Capitulo II. Caracterización del sistema.



Capitulo II. Caracterización del sistema.

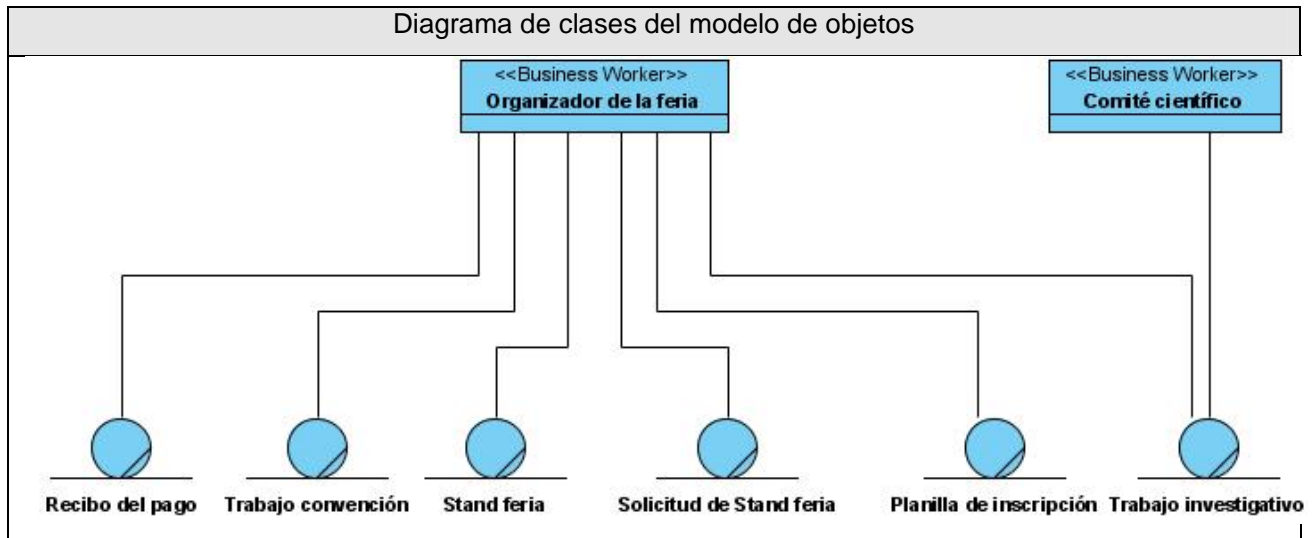


Capitulo II. Caracterización del sistema.



El modelo de objetos se utiliza para representar las relaciones entre los trabajadores del negocio y las entidades que están presentes en este.

Capítulo II. Caracterización del sistema.



2.2 Especificación de los requisitos de software.

Ivar Jacobson, Grady Booch y James Rumbaugh describen en su libro “El proceso unificado de desarrollo de software” la captura y especificación de requisitos como “el proceso de averiguar, normalmente en circunstancias difíciles, lo que se debe construir” con el propósito de “guiar el desarrollo hacia el sistema correcto” (JACOBSON *et al.* 2000).

Las ideas, necesidades o deseos de los clientes y miembros del equipo de desarrollo de lo que debe hacer el sistema, deben ser analizadas como requisitos candidatos. Los requisitos o requerimientos se pueden clasificar en funcionales o no funcionales.

Los requerimientos funcionales son condiciones o capacidades que el sistema debe cumplir. Los casos de usos del negocio sirven como punto de partida para identificar que debe hacer el sistema, ya que el sistema debe automatizar los procesos existentes para producir beneficios a los usuarios finales, así como brindar nuevas facilidades a los mismos.

Las cualidades o propiedades que el producto debe tener forman parte de los requerimientos no funcionales. Debe pensarse en estas propiedades como las características que hacen al producto atractivo, usable, rápido o confiable.

2.2.1 Requerimientos no funcionales

Los requisitos no funcionales se estructuraron en diversos aspectos. Se precisa que el sistema cumpla ciertas restricciones de **apariencia o interfaz externa** con el objetivo de lograr la mejor apariencia del producto, como:

Capítulo II. Caracterización del sistema.

- Todo el contenido publicado debe ser legible por los usuarios.
- Cada interfaz tiene que ser lo más sugerente posible, donde los usuarios puedan navegar sin necesidad de entrenamiento previo.
- Interfaz colorida y atractiva para los usuarios, que inspire deseos de participar.
- Interactivo, que los usuarios no lo vean como simples páginas que brindan información.
- Profesionalismo, las interfaces deben dotar de la respectiva seriedad.
- Se deben cumplir con los estándares de apariencia Web.
- La arquitectura de la información debe estar bien elaborada.

El sistema debe permitir un alto nivel de **facilidades de uso**, basado en el cumplimiento de los siguientes aspectos:

- Brindar facilidad de uso por personas que hablen otros idiomas además del español, en una primera versión el sistema debe permitir mostrarse en español e inglés, y aumentar la variedad de idiomas en las siguientes versiones.
- Accesibilidad para personas discapacitadas, permitiendo aumentar el tamaño de letra de los diferentes contenidos que se muestren.
- Se proporcionará a los usuarios el manual de usuario y toda la documentación necesaria para lograr un entendimiento total las acciones a realizar en el sistema.
- Los menús permitirán una navegación sencilla, tanto a los usuarios con conocimientos avanzados de informática como a los usuarios más inexpertos, esto se logrará a partir de una estructura de la información correcta, conjunta con las denominadas migajas de pan, que no son más que la información en todo momento a los usuarios del lugar donde se encuentra en el sistema.

Para lograr el mayor **rendimiento** posible, la aplicación se basará en el cumplimiento de ciertos pasos:

- Establecer como línea base la rapidez y eficiencia de las operaciones, tanto de acceso a datos como al mostrar la información.
- La información de cada una de las páginas no debe exceder de los 150 KB, para garantizar una carga rápida de estas por los navegadores Web, sobre todo los que se encuentren usando conexiones de red lentas.
- El sistema tiene que estar disponible las 24 horas del día, para garantizar la posibilidad de participación a los usuarios de distintas zonas horarias.

Capítulo II. Caracterización del sistema.

- La recuperación ante fallos tendrá que ocurrir lo más rápido posible, para garantizar que el evento se vea el menor tiempo afectado posible, con este fin se programarán la realización de copias de seguridad de la base de datos, en las horas en que menos accesos se realicen al sistema.
- Para lograr el mejor rendimiento y aprovechamiento de los recursos se destinará un servidor para el sistema gestor de base de datos y otro como servidor de aplicaciones.

Una vez concluido el proyecto se realizarán un grupo de acciones con el objetivo de asistir a los clientes, así como producir mejoras progresivas del sistema. Estas tareas de **soporte** incluyen:

- La realización de pruebas para determinar el grado de cumplimiento de los requerimientos de los clientes.
- Realización de mantenimientos de la base de datos.
- Entrega de las configuraciones del sistema, tanto digitales como impresas.
- Prestar en el momento que se solicite servicio de atención al cliente.
- Documentación necesaria para realizar una reinstalación del sistema.

El sistema debe cumplir determinados requisitos de **seguridad** para garantizar que no ocurran riesgos potenciales. La confiabilidad es uno de los puntos en los que el sistema debe centrarse, protegiendo la información de los usuarios, para que solo pueda ser accedida por las personas autorizadas. Otro aspecto en que estará enfocado, en lo que respecta a seguridad, es en la integridad de los datos, ya que se debe garantizar la entereza en toda la información que se almacene, bajo rigurosa protección contra alteraciones indebidas. Estos puntos serán manejados de manera tal que no afecten la disponibilidad de los usuarios autorizados al sistema y las prestaciones que este brinde.

Todas estas características serán garantizadas con la utilización de grupos de usuarios con roles definidos, de forma tal que se requiera autenticarse en el sistema para poder realizar ciertas acciones.

Para dar cumplimiento a este objetivo se establecen ciertas reglas, como son:

- La autenticación estará basada en el uso de nombres de usuario y contraseña.
- La contraseña tendrá como mínimo 7 caracteres y como máximo 32.
- Los usuarios estarán autorizados a realizar acciones según lo esté el rol al que pertenece.
- A los datos utilizados por el sistema se le deben hacer copias de seguridad con frecuencia determinada según la proximidad del evento, realizándose salvadas diarias en los días en que la convención se este realizando.

Capitulo II. Caracterización del sistema.

Entre los **requerimientos legales** que se deben tener en cuenta entran una serie de elementos, como lo son:

1- Condiciones Generales

La XIII edición de la Convención y Feria Internacional Informática 2009, en lo adelante y a los efectos de las presentes condiciones se denominará Informática 2009, ponen a disposición de los usuarios de forma gratuita el Sitio Web (en lo adelante Portal), dándole la condición de usuario del mismo a todo aquel que se identifique en el sistema y por tanto acepte las Condiciones Específicas que para la navegación se establecen.

2- Gratuidad en el servicio de acceso y utilización del Portal

Los servicios brindados en el Portal para los usuarios tienen carácter totalmente gratuito, independientemente de los servicios a que estos accedan.

3- Registro de Usuarios

El registro de los usuarios no es una condición indispensable para la navegación, pero necesaria para el acceso a algunos de los servicios que brinda el Portal, por lo que los usuarios que gozan de la condición de miembros, deben tener registrado sus datos personales y profesionales con el objetivo de facilitar el intercambio en caso que así **Informática 2009** lo entienda.

4- Uso apropiado del Portal y de los Servicios que en el mismo se brindan

El Usuario es el máximo responsable del uso que le da a la información adquirida en el Portal, así como de los servicios que el mismo brinda, manteniendo una estricta observancia de conformidad con la ley y demás normas éticas y morales reconocidas en Cuba e Internacionalmente. De igual forma se abstendrá de utilizar el servicio del Portal con fines ilícitos, contrarios a los intereses de INFORMÁTICA 2009, terceros y demás usuarios.

5- Adquisición de los Contenidos que conforman el Portal

Queda totalmente prohibida la adquisición de dibujos, gráficos, imágenes, sonidos, fotografías y demás herramientas que en materia de Software y Hardware sean utilizadas en la conformación y/o publicación del Portal, sin la conformidad expresa o tácita de INFORMÁTICA 2009 y terceros. Por lo que el Usuario está conminado a utilizar los contenidos del portal de forma correcta en total licitud.

6- Aplicación de la política anti-correo masivo en relación a los servicios que brinda el Portal

Queda prohibido a los usuarios la remisión o envío a través del Portal de cualquier tipo o clase de información con objetivos comerciales o publicitarios sin previo conocimiento y autorización de los administradores del Portal, así como a la utilización de listas o foros a las cuales se pueda acceder a través del mismo.

7- Utilización de Hiperenlaces a partir de la utilización del Portal

Capítulo II. Caracterización del sistema.

La utilización de hiperenlaces utilizando el Portal estará sujeta a las siguientes condiciones:

- La página Web en la que se establezca el hiperenlace no podrá tener contenidos ilícitos que contravengan la moral y las buenas costumbres.
- Con excepción de los signos que forman parte del hiperenlace, la página Web objeto del mismo no podrá tener ninguna marca comercial, eslogan, rótulo de establecimiento y demás signos distintivos perteneciente a INFORMÁTICA 2009.
- Con el Hiperenlace solo se permitirá el acceso al Portal, prohibiéndose su reproducción.
- El Hiperenlace será permitido exclusivamente para la página principal del Portal. <http://www.informaticahabana.com> o <http://www.informaticahabana.cu>
- El hiperenlace no creará en ningún caso relaciones entre el Portal y el o los propietarios de la Web.

8- Uso de Licencia

INFORMÁTICA 2009 no autoriza el uso del Portal en materias de propiedad industrial e intelectual así como otros derechos o prioridades protegidos por la ley.

9- Retiro del acceso al Portal y sus servicios

El incumplimiento de las condiciones aquí descritas por parte de los usuarios constituye una condición para que el INFORMÁTICA 2009 retire en cualquier momento y sin previo aviso, el acceso del Portal y sus servicios.

10- Vigencia

La vigencia del servicio de Portal así como otros Servicios tendrán como principio una duración ilimitada. No obstante INFORMÁTICA 2009 se reserva el derecho de dar por terminada o en su defecto suspender los servicios de Portal y demás asociados a este, sin perjuicio de lo que hubiere dispuesto al respecto en las presentes condiciones. Siempre que las circunstancias lo permitan, INFORMÁTICA 2009 advertirá previamente la terminación o suspensión de servicio de Portal y de los restantes Servicios.

11- Ley Aplicable y Jurisdicción

Las condiciones aquí recogidas se rigen por las regulaciones y leyes de la República de Cuba.

Para garantizar la **portabilidad** del sistema:

- Se debe diseñar para ser desarrollado sin basarse en características propias de algún sistema operativo, para lograr un producto multiplataforma, a pesar de esto, se recomienda realizar el despliegue del producto en servidores con tecnología Linux.

En lo relativo a **confiabilidad**:

Capítulo II. Caracterización del sistema.

- El sistema debe ser preciso en la información que le suministre al usuario para evitar provocar confusión de este.
- Debe estar bien documentado, para permitir minimizar el tiempo de mantenimiento en caso necesario.
- Debe contar el sistema con la funcionalidad de aviso de fallos a las personas involucradas con la aplicación, con el fin de solucionar los problemas en el menor tiempo posible.

Referido a la **interfaz interna**:

- El sistema requiere que algunos usuarios realicen pago electrónico, para lo que se utilizará una plataforma de pago brindada por la empresa Solways, que se utilizará con la inclusión en el sistema de un enlace a dicha plataforma.

Se debe brindar toda la **ayuda y documentación** posible:

- Documentando todas las tareas y operaciones que realice el software.
- Crear el glosario de términos.
- Debe poseer además los documentos de ayuda que garanticen el asesoramiento e información al usuario acerca de los procesos informatizados; donde quedarán reflejadas las funcionalidades del sistema y su manipulación.
- Se debe proporcionar el Manual de Usuario, donde se describa detalladamente las características y usos del software.

El **diseño y la implementación** estará regulado por exigencias como que:

- El sistema deberá ser implementado siguiendo un estándar de codificación propuesto por el líder de proyecto, el cual debe ser adjuntado a la documentación del software.
- Tanto la arquitectura como el diseño del software deben ser elaborados a partir del uso de patrones arquitectónicos y de diseño.
- Se requiere del uso del lenguaje PHP para la realización de este sistema.
- De dispone de un servidor de base de datos con el SGBD MySQL, por lo que la aplicación debe ser desarrollada para emplear este tipo de gestor de base de datos.

Los ordenadores que se utilicen para desplegar el sistema deben poder usar cualquier sistema operativo en el cual existan los siguientes **software**:

Capítulo II. Caracterización del sistema.

- Servidor de aplicaciones con intérprete de PHP 5.0.
- SGBD MySql 5.0.

En cuanto al **hardware** el sistema requiere de disponer de:

- Un servidor donde estará el SGBD MySql.
- Un servidor donde estará publicado el sistema mediante el servidor de aplicaciones.

2.2.2 Requerimientos funcionales.

Las funcionalidades u operaciones que el producto debe realizar forman parte de los requerimientos funcionales. Estas deben verse como los procesos a automatizar y las ventajas que tiene el producto sobre los procesos de negocio sin la utilización del mismo.

La satisfacción del cliente estará regulada por el grado de cumplimiento de los requisitos funcionales que a continuación se mencionan:

1. Registro en línea.
2. Gestión de contenidos
 - 2.1. Crear contenido
 - 2.2. Editar contenido
 - 2.3. Borrar contenido
 - 2.4. Ver contenidos publicados
3. Gestión de sesión de usuario
 - 3.1. Iniciar sesión
 - 3.2. Cerrar sesión
 - 3.3. Solicitar nueva contraseña
4. Gestión del perfil de usuario
 - 4.1. Configurar datos de usuario
 - 4.2. Configurar preferencias de participación
 - 4.3. Listar actividades de preferencia
5. Acceder a salón de eventos
 - 5.1. Listar eventos
 - 5.2. Ver ponencias presenciales
 - 5.3. Ver ponencias virtuales
 - 5.4. Ver relatorías
 - 5.5. Ver opiniones del evento

Capítulo II. Caracterización del sistema.

- 5.6. Ver las galerías de fotos
- 6. Acceder a las memorias
 - 6.1. Ver el texto general de cada edición
 - 6.2. Ver el salón de eventos
 - 6.3. Ver las opiniones
 - 6.4. Ver las relatorías
 - 6.5. Ver las galerías de fotos
 - 6.6. Ver los trabajos presentados
- 7. Envío de trabajos
- 8. Gestión de trabajos enviados
- 9. Gestión de noticias
 - 9.1. Crear noticia
 - 9.2. Editar noticia
 - 9.3. Borrar noticia
 - 9.4. Ver noticia publicada
- 10. Moderar preguntas y opiniones
- 11. Envío de opiniones o preguntas
- 12. Responder preguntas
- 13. Editar roles de usuario
- 14. Realización de pago electrónico
- 15. Administrar las diferentes ediciones realizadas
- 16. Gestión de la información de los diferentes eventos
- 17. Gestión de las actividades que se realizan en cada evento
- 18. Acceder a estadísticas
- 19. Realización de búsquedas
- 20. Gestión de configuraciones de los usuario
 - 20.1. Bloquear o desbloquear usuarios
 - 20.2. Editar estado del pago de los usuarios
- 21. Envío de solicitudes de stand expositivo
- 22. Gestión de solicitudes de stand expositivo

Capítulo II. Caracterización del sistema.

2.3 Modelo de caso de uso del sistema

2.3.1 Definición de los actores

Actores	Justificación
Usuario no autenticado	Usuario que no está identificado en el sistema, que puede registrarse en el sitio para crearse una cuenta de usuario, leer las noticias, opiniones y las diversas informaciones que estén publicadas, realizar búsquedas, y autenticarse en el sistema para disfrutar de otras funcionalidades.
Usuario autenticado	Este usuario puede realizar las mismas actividades que el usuario no autenticado, pero además puede acceder a los salones de eventos, a las memorias del evento, modificar su perfil de usuario, enviar preguntas u opiniones en las diferentes secciones, realizar solicitud de stand expositivo, enviar trabajo candidato a la convención, y terminar la sesión de usuario por la que es identificado en el sistema.
Editor	El editor hereda permisos del usuario autenticado pero en adición puede crear, modificar o borrar noticias.
Delegado virtual	Es un usuario autenticado que paga una cuota por ser un delegado del evento.
Ponente virtual	Delegado virtual que presenta un trabajo en la convención en la modalidad virtual y responde las diferentes preguntas que le realicen.
Gestor de estado de usuarios	Usuario autenticado que tiene privilegios para modificar el estado de los delegados, ya sea bloquearlos o desbloquearlos.
Organizador de la feria	El organizador de la feria es un usuario gestor de estado de los usuarios que puede además gestionar las solicitudes de los stands de la feria.
Comité científico	Encargado de gestionar los trabajos candidatos y determinar la forma de presentación de estos, puede además realizar todas las funciones de los gestores de estado de usuarios.
Administrador Web	Usuario que es gestor del estado de los usuarios pero que además puede gestionar la información que se publica, los roles de usuario, ver las estadísticas del sistema y también es encargado de gestionar el estado del pago de los delegados.
Moderador	Gestor de estado de los usuarios que modera las preguntas y opiniones que realizan los usuarios autenticados.

Capitulo II. Caracterización del sistema.

2.3.2 Listado de casos de uso.

CU - 1	Registrarse en el sitio
Actor	Usuario no autenticado
Descripción	Crear cuenta de usuario en el sistema
Referencia	R1

CU - 2	Ver noticia
Actor	Usuario no autenticado
Descripción	Visualizar el listado de noticias o una noticia en específico
Referencia	R8.4

CU - 3	Ver contenido
Actor	Usuario no autenticado
Descripción	Ver los diferentes contenidos publicados
Referencia	R2.4

CU - 4	Realizar búsqueda
Actor	Usuario no autenticado
Descripción	Realizar búsquedas de los contenidos publicados
Referencia	R19

CU - 5	Ver opiniones
Actor	Usuario no autenticado
Descripción	Leer las opiniones publicadas
Referencia	R5.5 y R6.3

CU - 6	Iniciar sesión
Actor	Usuario no autenticado
Descripción	Autenticarse en el sistema
Referencia	R3.1

CU - 7	Cerrar sesión
Actor	Usuario autenticado

Capítulo II. Caracterización del sistema.

Descripción	Terminar la sesión activa
Referencia	R3.2

CU - 8	Acceder a salón de eventos
Actor	Usuario autenticado
Descripción	Listar los diferentes eventos de la edición activa, y para cada uno de estos poder acceder a las diferentes actividades, como ponencias presenciales, ponencias virtuales, relatorías, galerías de foto y las opiniones del evento.
Referencia	R5.1, R5.2, R5.3,R5.4 y R5.6

CU - 9	Acceder a las memorias
Actor	Usuario autenticado
Descripción	Listar las diferentes ediciones efectuadas, y dentro de cada una de estas las actividades que se realizaron, como las relatorías, los trabajos presentados, las galerías de fotos, las opiniones y las descripciones generales de cada edición.
Referencia	R6.1, R6.2, R6.4, R6.5 y R6.6

CU - 10	Gestionar el perfil
Actor	Usuario autenticado
Descripción	Configurar los datos de usuario y preferencias de participación
Referencia	R4.1, R4.2 y R4.3

CU - 11	Enviar opinión o pregunta
Actor	Usuario autenticado
Descripción	Los usuarios autenticados pueden enviar preguntas u opiniones en distintas secciones del sistema.
Referencia	R11

CU - 12	Responder pregunta
Actor	Ponente virtual
Descripción	Los ponentes de la modalidad virtual deben responder las preguntas que le hagan los usuarios, una vez que estas hayan sido aprobadas por los moderadores.

Capítulo II. Caracterización del sistema.

Referencia	R12
CU - 13	Moderar preguntas y opiniones
Actor	Moderador
Descripción	Aprobar o rechazar la publicación de las preguntas y opiniones.
Referencia	R10
CU - 14	Enviar trabajo candidato
Actor	Usuario autenticado
Descripción	Las personas interesadas en participar en la convención pueden enviar trabajos con vistas a presentarlos en la misma mediante esta opción que lo envía directamente al comité científico correspondiente, y lo hace pasar por un flujo de trabajo definido.
Referencia	R7
CU - 15	Solicitar stand expositivo
Actor	Usuario autenticado
Descripción	Realizar solicitud de locales para presentar una investigación en la feria.
Referencia	R12.1
CU - 16	Gestionar noticias
Actor	Editor
Descripción	Los editores pueden crear, editar o borrar noticias, que son publicadas en el sistema y visibles por todos los usuarios.
Referencia	R9.1, R9.2 y R9.3
CU - 17	Realizar pago
Actor	Delegado virtual
Descripción	Los usuarios deben realizar el pago de una cuota por la participación en el evento, en las distintas modalidades, o en la feria; este pago le convierte en delegado virtual.
Referencia	R14

Capitulo II. Caracterización del sistema.

CU - 18	Gestionar estado de los usuarios
Actor	Gestor de estado de los usuarios
Descripción	Los usuarios que tengan estos privilegios pueden bloquear o desbloquear a los usuarios por realizar actividades indebidas en el sistema.
Referencia	R20.1

CU - 19	Gestionar solicitudes de stand expositivo
Actor	Organizador de la feria
Descripción	Los organizadores analizan las solicitudes de stand para exponer en la feria y las aprueban o no en dependencia de los intereses de la edición.
Referencia	R12.2

CU - 20	Gestionar trabajos candidatos
Actor	Comité científico
Descripción	Cada comisión se encarga de revisar los trabajos propuestos y determinar cuales serán presentados y en que modalidad.
Referencia	R8

CU - 21	Gestionar ediciones
Actor	Administrador Web
Descripción	El administrador puede gestionar las distintas ediciones que tiene el sistema, creando, modificando o borrando estas.
Referencia	R15

CU - 22	Gestionar eventos en cada edición
Actor	Administrador Web
Descripción	En cada edición que esté registrada en el sistema se pueden crear, editar o borrar los eventos que existen dentro de la misma.
Referencia	R16

CU - 23	Gestionar actividades dentro de evento
Actor	Administrador Web
Descripción	Dentro de cada evento se crean, editan o borran las diferentes actividades que

Capítulo II. Caracterización del sistema.

	se desarrollarán, ya sean ponencias virtuales, ponencias presenciales, relatorías, o galerías de foto.
Referencia	R17

CU - 24	Gestionar contenidos
Actor	Administrador Web
Descripción	Crear, editar o eliminar contenidos.
Referencia	R2.1, R2.2 y R2.3

CU - 25	Gestionar los roles de usuario
Actor	Administrador Web
Descripción	El administrador Web definirá los privilegios de cada usuario.
Referencia	R13

CU - 26	Acceder a estadísticas
Actor	Administrador Web
Descripción	Se visualizarán las estadísticas del sistema
Referencia	R18

CU - 27	Gestionar el pago de los usuarios
Actor	Administrador Web
Descripción	Actualización en el sistema de los usuarios que han pagado mediante la plataforma de pago
Referencia	R20.2

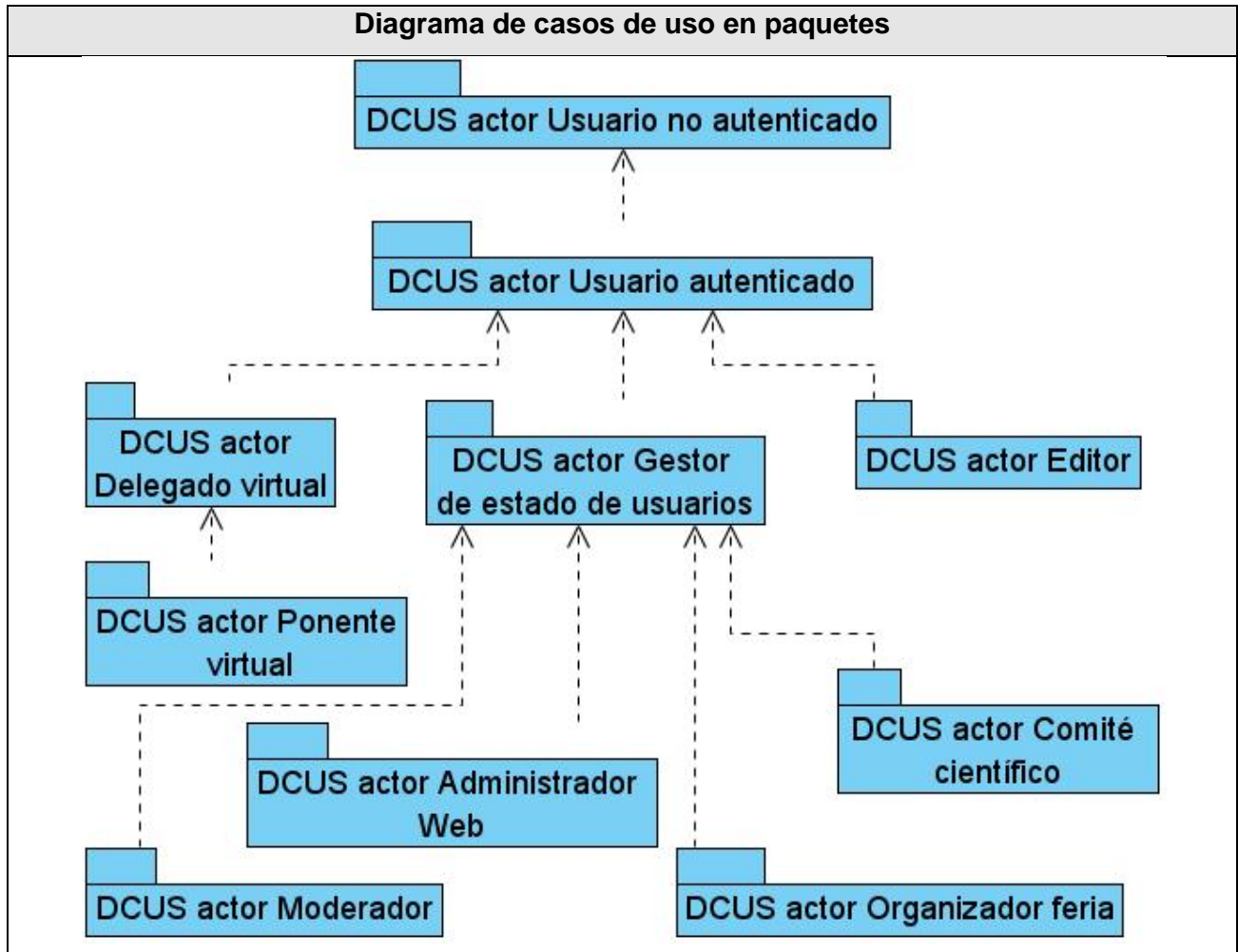
CU - 28	Solicitar nueva contraseña
Actor	Usuario no autenticado
Descripción	Solicitud para crear una nueva contraseña por el olvido de la actual
Referencia	R3.3

2.3.3 Diagrama de casos de uso del sistema.

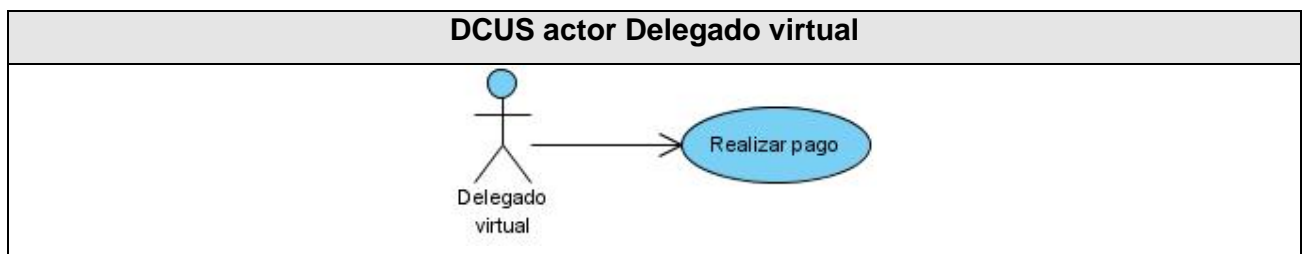
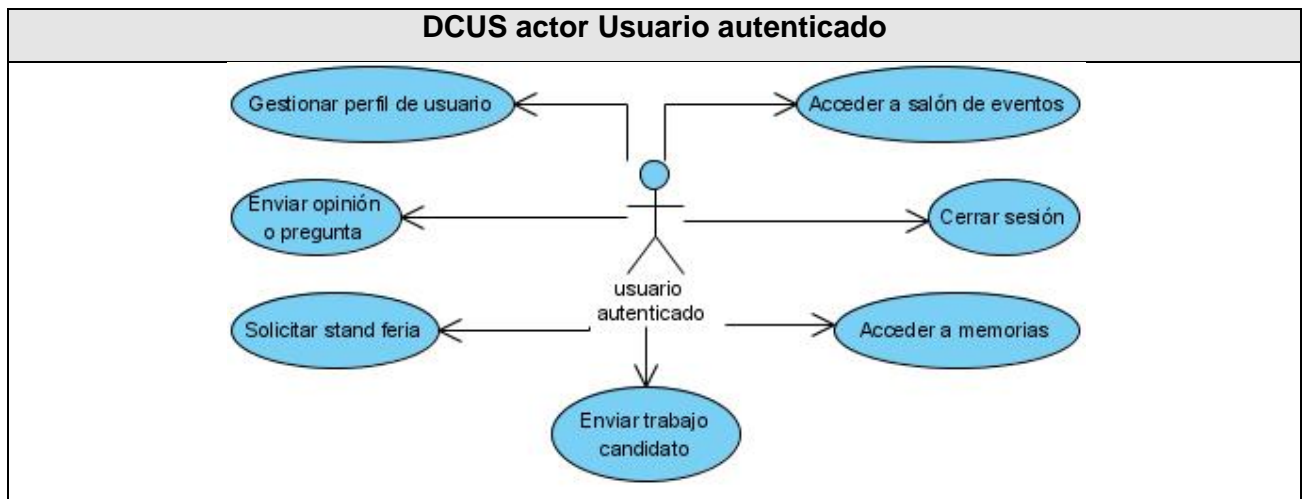
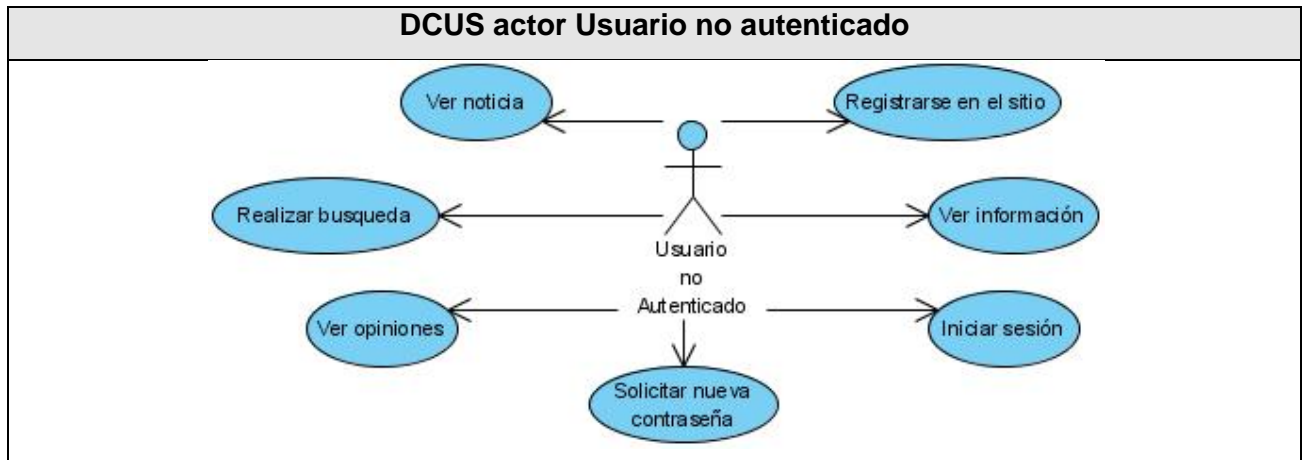
Para lograr una mejor visualización del diagrama de casos de uso, este se estructuró en paquetes, los cuales están relacionados, para representar las relaciones de herencia entre los actores incluidos en

Capitulo II. Caracterización del sistema.

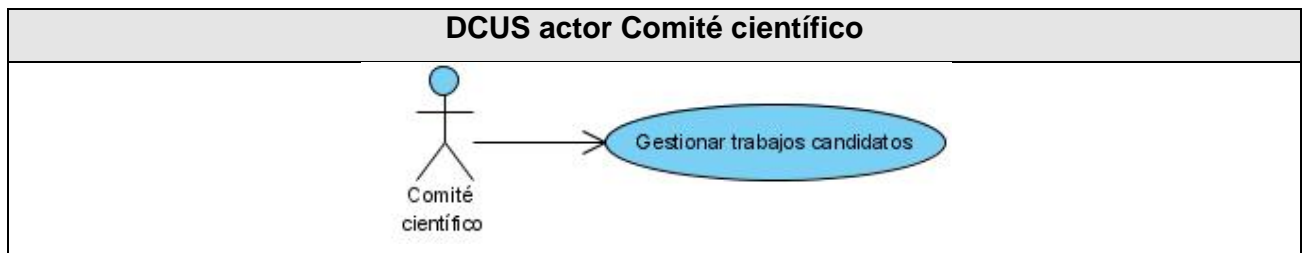
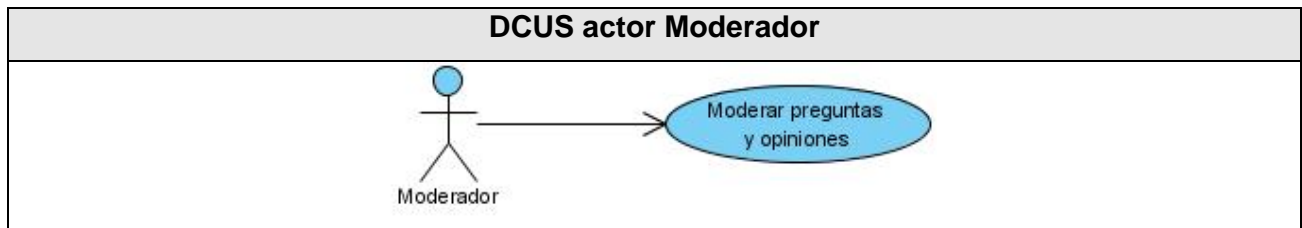
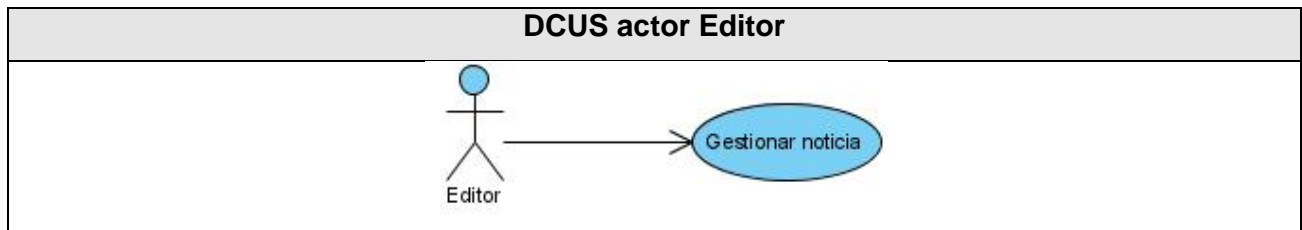
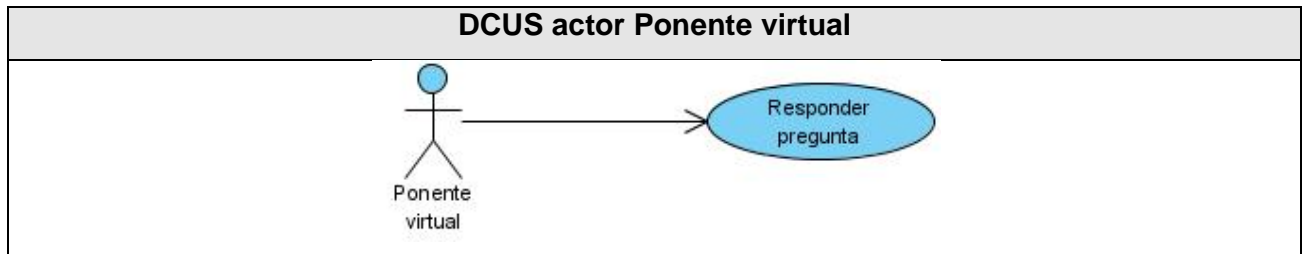
los diferentes paquetes. Cada paquete está conformado por un actor, y los casos de uso que este inicia.



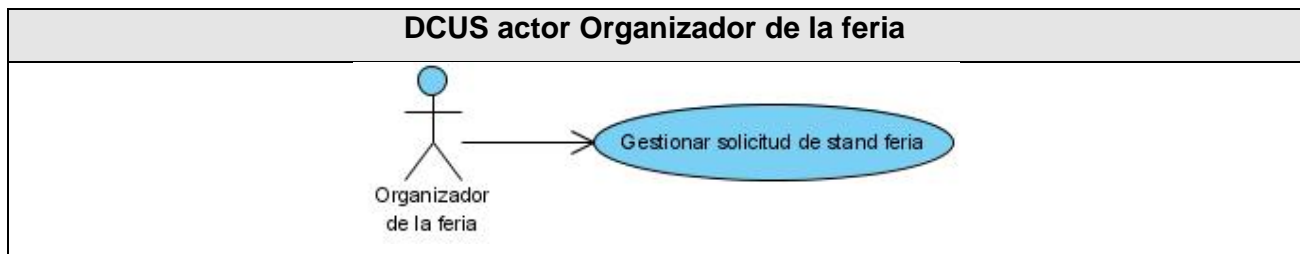
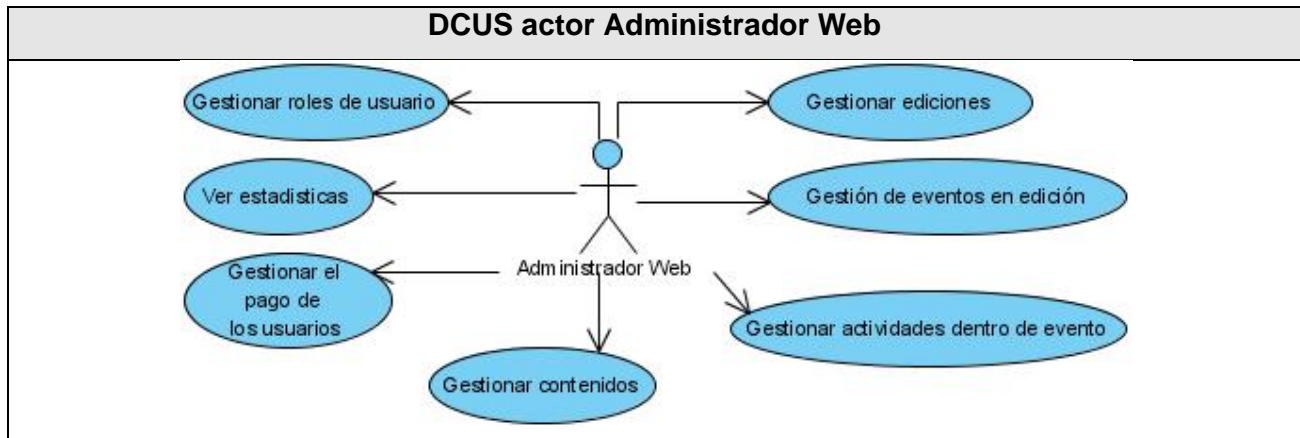
Capitulo II. Caracterización del sistema.



Capitulo II. Caracterización del sistema.



Capitulo II. Caracterización del sistema.



2.3.4 Casos de uso expandidos.

Caso de uso	
CU - 1	Registrarse en el sistema
Propósito	Que los usuarios se puedan crear cuentas de usuario en el sistema.
Actores: Usuario no autenticado	
Resumen: Crear cuenta de usuario en el sistema	
Referencias	R1
Acción del actor	Respuesta del sistema
1 El usuario de Internet accede a la página de registro y llena el formulario requerido y acciona el botón enviar.	2 El sistema valida los datos.
4 Recibe el correo, lee la contraseña generada e introduce usuario y contraseña.	3 Almacena los datos de usuario y envía un correo electrónico con la contraseña generada para garantizar la autenticidad del correo electrónico introducido.
6 Introduce la contraseña deseada.	5 Inicia la sesión del usuario y solicita una contraseña personalizada.
	7 Almacena la nueva contraseña de usuario.

Capítulo II. Caracterización del sistema.

Flujo alternativo 1	
Acción del actor	Respuesta del sistema
	2.1 Los datos son incorrectos, se informa al usuario para que los corrija.

Caso de uso	
CU - 2	Ver noticias
Propósito	Que los usuarios se mantengan informados de todo lo que ocurre sobre la convención y feria internacional de informática
Actores: Usuario no autenticado	
Resumen: Visualizar el listado de noticias y mostrar la seleccionada	
Referencias	R8.4
Acción del actor	Respuesta del sistema
1 El usuario visualiza en la portada del sistema las noticias por orden cronológico y acciona la que desee ver el artículo completo.	2 Obtiene los datos de la noticia y la muestra completamente.

Caso de uso	
CU - 3	Ver contenido
Propósito	Que los usuarios pueden tener toda la información relativa a la Convención y Feria Internacional de Informática
Actores: Usuario no autenticado	
Resumen: Ver los diferentes contenidos publicados	
Referencias	R2.4
Acción del actor	Respuesta del sistema
1 El usuario acciona en los menús el contenido que desea ver.	2 Obtiene los datos del contenido y lo muestra detalladamente.

Caso de uso	
CU - 4	Realizar búsqueda
Propósito	Qué el usuario pueda encontrar rápidamente la información que desea
Actores: Usuario no autenticado	

Capítulo II. Caracterización del sistema.

Resumen: Realizar búsquedas de los contenidos publicados y noticias	
Referencias	R19
Acción del actor	Respuesta del sistema
1 El usuario accede a la página de búsquedas y escribe las palabras deseadas.	2 Buscan las coincidencias con informaciones y noticias. 3 Muestra al usuario el resultado de la búsqueda.

Caso de uso	
CU - 5	Ver opiniones
Propósito	Que el usuario pueda conocer las valoraciones de otros usuarios.
Actores: Usuario no autenticado, moderador	
Resumen: Leer las opiniones publicadas	
Referencias	R5.5 y R6.3
Flujo normal 1	
Acción del actor	Respuesta del sistema
1 El usuario accede a las opiniones del evento mediante el menú.	2 Muestra las opiniones existentes.

Caso de uso	
CU - 6	Iniciar sesión
Propósito	Identificar al usuario
Actores: Usuario no autenticado	
Resumen: Autenticarse en el sistema	
Referencias	R3.1
Acción del actor	Respuesta del sistema
1 Escribe su nombre de usuario y contraseña en el formulario de iniciar sesión.	2 Verifica el nombre de usuario y contraseña. 3 Al comprobar que los datos son correctos inicia la sesión del usuario. 4 Almacena en la sesión los datos del usuario.
Flujo alternativo	
Acción del actor	Respuesta del sistema

Capítulo II. Caracterización del sistema.

	2.1 Los datos no son correcto y se vuelven a solicitar al usuario, en caso de cometer el mismo error en tres ocasiones se comienza a solicitar además el texto de una imagen captcha.
--	---

Caso de uso	
CU - 7	Cerrar sesión
Propósito	Terminar la sesión de usuario activa
Actores: Usuario autenticado	
Resumen: Terminar la sesión activa	
Referencias	R3.2
Acción del actor	Respuesta del sistema
1 El usuario selecciona en el menú la opción de cerrar sesión.	2 Cierra la sesión de usuario.

Caso de uso	
CU - 8	Acceder a salón de eventos
Propósito	Que los usuarios puedan acceder a las diferentes actividades de cada evento
Actores: Usuario autenticado	
Resumen: Listar los diferentes eventos de la edición activa, y para cada uno de estos poder acceder a las diferentes actividades, como ponencias presenciales, ponencias virtuales, relatorías, galerías de fotos y las opiniones del evento.	
Referencias	R5.1, R5.2, R5.3,R5.4 y R5.6
Acción del actor	Respuesta del sistema
1 El usuario accede al salón de eventos donde se le listan todos los eventos y selecciona evento al que desea acceder.	2 Muestra la descripción del evento y los enlaces a los distintos tipos de actividades que se realizan dentro del evento seleccionado.
3 Selecciona una actividad.	
Escenario 1	
Acción del actor	Respuesta del sistema

Capítulo II. Caracterización del sistema.

3.1 Usuario accede a las ponencias presenciales 3.3 Selecciona la ponencia presencial de su interés.	3.2 Lista las ponencias presenciales. 3.4 Muestra los datos generales del trabajo y se brinda la posibilidad de descargarlo.
Escenario 2	
Acción del actor	Respuesta del sistema
3.1 Usuario accede a las ponencias virtuales. 3.3 Selecciona la ponencia virtual de su interés.	3.2 Lista las ponencias virtuales. 3.4 Muestra los datos generales del trabajo y se brinda la posibilidad de descargarlo. 3.5 Brinda la posibilidad de enviar una pregunta.
Escenario 3	
Acción del actor	Respuesta del sistema
3.1 Usuario accede a las relatorías. 3.3 Selecciona la relatoría de su interés.	3.2 Listan las relatorías. 3.4 Muestra el texto perteneciente a la relatoría seleccionada.
Escenario 4	
Acción del actor	Respuesta del sistema
3.1 Usuario accede a la galería de fotos. 3.3 Selecciona la galería de foto que desea ver.	3.2 Listan las distintas galerías de fotos existentes del evento. 3.4 Muestra la galería de fotos seleccionada.
Puntos de extensión	
CU – 11	

Caso de uso	
CU - 9	Acceder a memorias
Propósito	Que los usuarios puedan ver lo que ha ocurrido en encuentros anteriores
Actores: Usuario autenticado	
Resumen: Listar las diferentes ediciones efectuadas, y dentro de cada una de estas las actividades que se realizaron, como las relatorías, los trabajos presentados, las galerías de fotos, las opiniones y las descripciones generales de cada edición.	
Referencias	R6.1, R6.2, R6.3, R6.4, R6.5 y R6.6

Capítulo II. Caracterización del sistema.

Acción del actor	Respuesta del sistema
1 El usuario accede mediante el menú a las memorias. 3 Selecciona del listado la edición de interés. 5 Selecciona el evento de interés. 7 Selecciona el tipo de actividad de interés.	2 Listan las distintas ediciones que se han realizado, con un texto como descripción general de lo que paso en cada cita. 4 Lista el salón de eventos que tuvo lugar dentro de la edición. 6 Muestra una descripción del evento y lista los tipo de actividades realizadas en él.
Escenario 1	
Acción del actor	Respuesta del sistema
7.1 Selecciona ver las opiniones del evento.	7.2 Muestra las opiniones del evento.
Escenario 2	
Acción del actor	Respuesta del sistema
7.1 Selecciona ver las relatorías. 7.3 Selecciona la relatoría que desea acceder.	7.2 Muestra las distintas relatorías existentes. 7.4 Muestra el texto perteneciente a la relatoría seleccionada.
Escenario 3	
Acción del actor	Respuesta del sistema
7.1 Selecciona ver las galerías de fotos. 7.3 Selecciona la galería de fotos que desee ver.	7.2 Lista las distintas galerías de fotos. 7.4 Muestra la galería de foto seleccionada.
Flujo alternativo 4	
Acción del actor	Respuesta del sistema
7.1 Selecciona ver los trabajos presentados. 7.3 Descarga mediante el enlace, el trabajo que desea ver.	7.2 Lista los trabajos presentados en el evento seleccionado con sus respectivos enlaces para descargarlos.

Caso de uso	
CU - 10	Gestionar el perfil
Propósito	Que el usuario pueda configurar sus datos y acceder a sus intereses de participación de forma rápida

Capítulo II. Caracterización del sistema.

Actores: Usuario autenticado	
Resumen: Configurar los datos de usuario y preferencias de participación	
Referencias	R4.1, R4.2 y R4.3
Escenario 1	
Acción del actor	Respuesta del sistema
1 El usuario desea modificar sus datos personales y accede a la opción prevista para este fin. 3 Modifica los datos de usuario y los envía.	2 Muestran los datos actuales permitiéndole modificarlos. 4 Almacena los datos.
Escenario 2	
Acción del actor	Respuesta del sistema
1 El usuario desea modificar sus preferencias de participación y accede a la opción creada con este objetivo. 3 Selecciona las preferencias de participación que desea borrar.	2 Muestran las preferencias actuales de participación junto a un enlace para borrar la preferencia. 4 Borra las preferencias seleccionadas.
Escenario 3	
Acción del actor	Respuesta del sistema
1 El usuario desea acceder de forma rápida a las actividades de su interés, por lo que selecciona directamente el enlace de la misma colocado en el cuadro del perfil.	2 Muestra la actividad solicitada.

Caso de uso	
CU – 11	Enviar opinión o pregunta
Propósito	Favorecer el dialogo científico
Actores: Usuario autenticado	
Resumen: Los usuarios autenticados pueden enviar preguntas u opiniones en distintas secciones del sistema.	
Referencias	R11
Acción del actor	Respuesta del sistema
1 Escribe la pregunta u opinión y la envía a	2 Almacena la pregunta u opinión.

Capítulo II. Caracterización del sistema.

través del formulario creado con este fin.

Caso de uso	
CU – 12	Responder pregunta
Propósito	Favorecer el dialogo científico
Actores: Ponente virtual	
Resumen: Una vez que las preguntas son aprobadas por los moderadores, cada ponente virtual es encargado de darle respuesta a las que estén relacionadas al trabajo que presente	
Referencias	R12
Acción del actor	Respuesta del sistema
1 Accede a la página destinada a su trabajo. 3 Selecciona la pregunta a la que va a dar respuesta. 5 Se escribe la respuesta y se envía.	2 Muestra la página donde está publicado su trabajo, y en ella tendrá habilitado la opción de responder preguntas. 4 Genera un formulario para escribir la respuesta a la pregunta seleccionada. 6 Almacena la respuesta.

Caso de uso	
CU – 13	Moderar preguntas y opiniones
Propósito	Evitar la publicación de preguntas u opiniones indebidas o repetidas
Actores: Moderador	
Resumen: Modera las diferentes preguntas que se vayan haciendo en el momento que se está realizando una ponencia virtual, así como las distintas opiniones emitidas.	
Referencias	R10
Acción del actor	Respuesta del sistema
1 Accede a la página creada para listar las preguntas u opiniones en los distintos estados. 3 Analiza la pregunta u opinión. 4 Determina el estado de la pregunta u opinión y selecciona almacenar los cambios.	2 Lista las preguntas u opiniones con los datos del usuario que la realizó, la fecha y hora, lugar, y la pregunta como tal, filtradas por el estado de estas. 5 Actualiza el estado de la pregunta u opinión.

Capítulo II. Caracterización del sistema.

Caso de uso	
CU – 14	Enviar trabajo candidato
Propósito	Que los delegados puedan presentar trabajos en la convención
Actores: Usuario autenticado	
Resumen: Las personas interesadas en participar en la convención pueden enviar trabajos con vistas a presentarlos en la misma mediante esta opción que lo envía directamente al comité científico correspondiente, y lo hace pasar por un flujo de trabajo definido.	
Referencias	R7
Acción del actor	Respuesta del sistema
1 Accede al formulario de envío de trabajos 3 Llena los campos para el envío del trabajo, donde van una breve descripción del trabajo, comisión a la que envía, la modalidad de presentación deseada, el documento del trabajo y otros datos del mismo. 6 Recibe la notificación del comité científico. 7 El trabajo fue aprobado por el comité científico, y producto de esto recibe una notificación sobre el pago de la cuota requerida en caso que no la halla pagado aún.	2 Muestra el formulario requerido para enviar un trabajo. 4 Almacena el trabajo. 5 La comisión correspondiente determina sobre la solicitud.
Flujo alternativo 1	
Acción del actor	Respuesta del sistema
6.1 El trabajo fue rechazado por el comité científico.	
Puntos de extensión.	
CU – 20	

Caso de uso	
CU – 15	Solicitar stand expositivo
Propósito	Que los expositores y los organizadores del evento logren un mejor entendimiento sobre las solicitudes de stand expositivo

Capítulo II. Caracterización del sistema.

Actores: Usuario autenticado	
Resumen: Realizar solicitud de locales para presentar una investigación en la feria.	
Referencias	R21
Acción del actor	Respuesta del sistema
<p>1 El usuario desea realizar una solicitud de un stand expositivo, y para ello accede a esta opción del menú.</p> <p>3 Selecciona el área deseada y llena los datos del formulario.</p> <p>6 Recibe la notificación sobre el resultado de la solicitud.</p> <p>7 La solicitud fue aprobada, y producto de esto recibe una notificación sobre el pago de la cuota requerida, según las condiciones del área solicitada.</p>	<p>2 Muestra un mapa del recinto ferial con las diferentes secciones disponibles, junto a un formulario para que el usuario realice la solicitud del área deseada, describiendo los argumentos de dicha solicitud.</p> <p>4 Almacena la solicitud.</p> <p>5 Al detectar un cambio en el estado de la solicitud notifica al usuario.</p>
Flujo alternativo	
Acción del actor	Respuesta del sistema
6.1 Recibe una notificación de que la solicitud fue rechazada.	

Caso de uso	
CU – 16	Gestionar noticias
Propósito	Mantener a los usuarios informados
Actores: Editor	
Resumen: Los editores pueden crear, editar o borrar noticias.	
Referencias	R9.1, R9.2 y R9.3
Escenario 1	
Acción del actor	Respuesta del sistema
<p>1 El editor accede a la página de crear noticia.</p> <p>3 Llena los campos de la noticia y se envían.</p>	<p>2 Muestra la página requerida.</p> <p>4 Almacena la noticia.</p>

Capitulo II. Caracterización del sistema.

	5 Publica la noticia.
Escenario 2	
Acción del actor	Respuesta del sistema
1 El editor accede a editar la noticia.	2 Muestra la información de la noticia y se brinda la posibilidad de editarla.
3 Edita la noticia y la envía.	4 Actualiza la noticia.
Escenario 3	
Acción del actor	Respuesta del sistema
1 El editor accede a la noticia.	2 Muestra la información de la noticia y brinda una opción para borrarla.
3 Selecciona borrar la noticia.	4 Borra la noticia.

Caso de uso	
CU – 17	Realizar pago
Propósito	Pagar la cuota requerida
Actores: Delegado virtual	
Resumen: Los usuarios deben realizar el pago de una cuota por la participación en el evento, en las distintas modalidades, o en la feria; este pago le convierte en delegado virtual.	
Referencias	R14
Flujo normal 1	
Acción del actor	Respuesta del sistema
1 Selecciona la opción de realizar pago.	2 Provee la plataforma de pago electrónico.

Caso de uso	
CU - 18	Gestionar el estado de los usuarios
Propósito	Bloquear o desbloquear usuarios para prevenir posibles problema con usuarios maliciosos
Actores: Gestor de estado de usuarios	
Resumen: Los usuarios que tengan estos privilegios pueden bloquear o desbloquear a los usuarios por realizar actividades indebidas en el sistema.	
Referencias	R20.1
Escenario 1	

Capitulo II. Caracterización del sistema.

Acción del actor	Respuesta del sistema
<p>1 Detecta algún comportamiento malicioso en alguna solicitud de stand expositivo, en el envío de trabajos candidatos a la convención, o utilización de expresiones irrespetuosas o agresivas en la realización de preguntas u opiniones.</p> <p>2 Accede a la página destinada a editar el estado de los usuarios.</p> <p>4 Localiza al usuario y llena el formulario del motivo del bloqueo, y lo envía.</p>	<p>3 Muestra el formulario previsto para tal fin.</p> <p>5 Actualiza el estado del usuario.</p> <p>6 Almacena el cambio de estado, con el usuario que bloquea, el bloqueado, la causa, la hora, la fecha, y tiempo por el que se bloqueó.</p> <p>7 Solo se permite el acceso del usuario bloqueado a visualizar las informaciones, noticias y el envío de opiniones por si desea plantear que es una equivocación, informándole el motivo de su bloqueo.</p>
Escenario 2	
Acción del actor	Respuesta del sistema
<p>1 Determina que es preciso desbloquear un usuario, producto de alguna equivocación u otro motivo.</p> <p>2 Accede a la página destinada a editar el estado de los usuarios.</p> <p>4 Localiza al usuario y llena el formulario del motivo del desbloqueo, y lo envía.</p>	<p>3 Muestra el formulario previsto para tal fin.</p> <p>5 Actualiza el estado del usuario.</p> <p>6 Almacena el cambio de estado, con el usuario que desbloquea, el desbloqueado, la causa, la hora y la fecha.</p> <p>7 Permite el acceso del usuario a todas las secciones de los usuarios autenticados.</p>

Caso de uso

Capítulo II. Caracterización del sistema.

CU – 19	Gestionar solicitud de expositor feria	
Propósito	Decidir las solicitudes de stand que deben ser aprobadas.	
Actores: Organizador de la feria		
Resumen: Los organizadores analizan las solicitudes de stand para exponer en la feria y las aprueban o no en dependencia de los intereses de la edición.		
Referencias	R22	
Acción del actor	Respuesta del sistema	
1 Accede al módulo designado para analizar las solicitudes de stand expositivo. 3 Selecciona la solicitud a analizar. 5 Analiza la solicitud. 6 Aprueba la solicitud.	2 Lista las solicitudes realizadas y en espera de aprobación. 4 Muestra los datos de la solicitud seleccionada. 7 Actualiza el estado de la solicitud. 8 Informa al delegado solicitante de la aprobación de su solicitud y del pago de la cuota requerida.	
Flujo alternativo		
Acción del actor	Respuesta del sistema	
5.1 Rechaza la solicitud emitiendo una explicación.	5.2 Actualiza el estado de la solicitud. 5.3 Informa al delegado solicitante de la desaprobación de su solicitud.	
Puntos de extensión.		
CU – 15		

Caso de uso		
CU - 20	Gestionar trabajos candidatos	
Propósito	Definir los trabajos que se van a presentar en cada comisión de la convención	
Actores: Comité científico de trabajo		
Resumen: Cada comisión se encarga de revisar los trabajos propuestos y determinar cuales serán presentados y en que modalidad de presentación.		
Referencias	R8	
Acción del actor	Respuesta del sistema	
1 Accede a la página de gestionar los trabajos	2 Se listan todos los trabajos que están es espera	

Capítulo II. Caracterización del sistema.

<p>presentados.</p> <p>3 Selecciona el trabajo que desee analizar.</p> <p>5 Analiza el trabajo.</p> <p>6 Determina que el trabajo puede ser aprobado y marca esta opción.</p> <p>7 Determina la forma de presentación del trabajo y envía la información.</p>	<p>de ser aprobados o rechazados.</p> <p>4 Se muestran los detalles del trabajo, y un enlace para descargarlo.</p> <p>8 Actualiza el estado de la solicitud.</p> <p>9 Actualiza la lista de trabajos aprobados.</p>
Flujo alternativo 1	
Acción del actor	Respuesta del sistema
<p>5.1 Determina que pertenece a otra comisión.</p> <p>5.2 Envía el trabajo al comité científico correcto.</p>	5.3 Actualiza el estado de la solicitud.
Flujo alternativo 2	
Acción del actor	Respuesta del sistema
6 Rechaza la solicitud de presentación del trabajo y selecciona almacenar el cambio de estado de la solicitud.	6.1 Actualiza el estado de la solicitud.
Puntos de extensión	
CU – 14	

Caso de uso	
CU – 21	Gestionar ediciones
Propósito	Almacenar en el sistema la información de todos los eventos realizados, logrando la reutilización del mismo en cada edición
Actores: Administrador Web	
Resumen: El administrador puede gestionar las distintas ediciones que tiene el sistema, creando, modificando o borrando estas.	
Referencias	R15
Flujo normal 1	
Escenario 1	Respuesta del sistema
1 Selecciona crear edición.	2 Muestra el formulario con los datos que se

Capítulo II. Caracterización del sistema.

3 Introduce los datos correspondientes a la edición que se desea crear y se envían.	requieren. 4 Actualiza el listado de ediciones almacenadas.
Escenario 2	
Acción del actor	Respuesta del sistema
1 Selecciona editar edición. 3 Selecciona la edición que se desea modificar. 5 Modifica los datos de la edición y se envían.	2 Lista las distintas ediciones almacenadas. 4 Muestran los datos de la edición y brinda la posibilidad de modificarlos. 6 Se actualiza el listado de ediciones almacenadas.
Escenario 3	
Acción del actor	Respuesta del sistema
1 Selecciona borrar edición. 3 Selecciona la cita que se desea borrar. 5 Acciona la opción de borrar la edición.	2 Lista las distintas ediciones almacenadas. 4 Muestra los datos de la edición y se brinda la opción de borrarla. 6 Actualiza el listado de ediciones almacenadas, y se eliminan todos los datos vinculados a la edición eliminada.

Caso de uso	
CU – 22	Gestionar eventos en cada edición
Propósito	Controlar los distintos eventos que se realicen por edición
Actores: Administrador Web	
Resumen: En cada edición que esté registrada en el sistema se pueden crear, editar o borrar los eventos que existen dentro de la misma.	
Referencias	R16
Escenario 1	
Acción del actor	Respuesta del sistema
1 Selecciona crear evento dentro de la edición. 3 Introduce los datos correspondientes al evento que se desea crear y se envían.	2 Muestra el formulario con los datos que se requieren. 4 Actualiza el listado de eventos almacenadas.
Escenario 2	
Acción del actor	Respuesta del sistema

Capítulo II. Caracterización del sistema.

1 Selecciona editar evento.	2 Lista los distintos eventos almacenados con la edición a la que pertenecen.
3 Selecciona el evento que se desea modificar.	4 Muestran los datos del evento y se da la posibilidad de editarlos.
5 Modifica los datos del evento y se envían.	6 Actualiza los datos del eventos.
Escenario 3	
Acción del actor	Respuesta del sistema
1 Selecciona borrar eventos.	2 Lista los distintos eventos almacenados con la edición a la que pertenecen.
3 Selecciona el evento que desea borrar.	4 Muestra los datos del evento y se da la opción de borrar.
5 Acciona la opción de borrar el evento.	6 Actualiza la lista de eventos almacenados y datos relacionados.

Caso de uso	
CU – 23	Gestionar actividades dentro de evento
Propósito	Administrar las distintas actividades que se realizan dentro de cada evento
Actores: Administrador Web	
Resumen: Dentro de cada evento se crean, editan o borran las diferentes actividades que se desarrollarán, ponencias virtuales, ponencias presenciales, relatorías o galerías de todos.	
Referencias	R17
Escenario 1	
Acción del actor	Respuesta del sistema
1 Selecciona crear actividad.	2 Muestra el formulario con los datos que se requieren.
3 Introduce los datos correspondientes a la actividad que se desea crear y se envían.	4 Actualiza el listado de actividades almacenadas.
Escenario 2	
Acción del actor	Respuesta del sistema
1 Selecciona editar actividad.	2 Lista las distintas actividades almacenadas con la edición y evento al que pertenece.
3 Selecciona la actividad que se desea modificar.	4 Muestra los datos de la actividad y se da la

Capítulo II. Caracterización del sistema.

5 Modifica los datos de la actividad y se envían.	posibilidad de editarlos. 6 Actualiza los datos de la actividad.
Escenario 3	
Acción del actor	Respuesta del sistema
1 Selecciona borrar actividad.	2 Lista las distintas actividades almacenadas con la edición y evento al que pertenece.
3 Selecciona la actividad que se desea borrar.	4 Muestra los datos de la actividad y se da la opción de borrarla.
5 Selecciona la opción de borrar la actividad.	6 Actualiza la lista de actividades del evento y se eliminan todos los datos relacionados con la actividad.

Caso de uso	
CU – 24	Gestionar contenidos
Propósito	Administrar los distintos contenidos que se deseen publicar en el sistema
Actores: Administrador Web	
Resumen: Crear, editar o eliminar contenidos de información para el usuario	
Referencias	R2.1, R2.2 y R2.3
Escenario 1	
Acción del actor	Respuesta del sistema
1 Selecciona crear contenido.	2 Muestra el formulario con los datos que se requieren.
3 Se introducen los datos correspondientes al contenido que se desea crear y se envían.	4 Almacena el contenido.
Escenario 2	
Acción del actor	Respuesta del sistema
1 Selecciona editar contenido.	2 Lista los distintos contenidos almacenados.
3 Selecciona el contenido que se desea modificar.	4 Muestra los datos del contenido y se da la posibilidad de editarlos.
5 Modifica los datos del contenido y se envían.	6 Actualiza los datos del contenido.
Escenario 3	

Capítulo II. Caracterización del sistema.

Acción del actor	Respuesta del sistema
1 Selecciona borrar contenido.	2 Lista los distintos contenidos almacenados.
3 Selecciona el contenido que se desea borrar.	4 Muestran los datos del contenido y se da la opción de borrarlo.
5 Selecciona la opción de borrar contenido.	6 Actualiza la lista de contenidos.

Caso de uso	
CU – 25	Gestionar los roles de usuario
Propósito	Establecer los distintos permisos necesarios
Actores: Administrador Web	
Resumen: El administrador Web definirá los privilegios de cada usuario.	
Referencias	R13
Escenario 1	
Acción del actor	Respuesta del sistema
1 Accede al listado total de usuarios.	2 Lista los usuarios registrados, junto a filtros para posibilitar localizar rápidamente a los usuarios deseados.
3 Selecciona el usuario al que desea modificar los roles de que tiene en el sistema.	4 Se obtienen los datos del usuario seleccionado incluidos los roles del mismo, y muestra todos estos datos.
5 Modifica los roles a los que pertenece el usuario en cuestión y al terminar acciona el botón de enviar.	6 Actualiza los roles de usuario del usuario seleccionado.

Caso de uso	
CU – 26	Acceder a estadísticas
Propósito	Conocer en todo momento las estadísticas de acceso al sistema desde Internet
Actores: Administrador Web	
Resumen: Visualizar las estadísticas del sistema	
Referencias	R18
Acción del actor	Respuesta del sistema
1 Selecciona la página de estadísticas del sistema.	2 Lista las estadísticas que el sistema controla.

Capitulo II. Caracterización del sistema.

Caso de uso	
CU - 27	Gestionar el pago de los usuarios
Propósito	Tener la relación de delegados que han efectuado el pago
Actores: Administrador Web	
Resumen: Actualizar en el sistema el estado de los pagos efectuados por los delegados	
Referencias	R20.2
Escenario 1	
Acción del actor	Respuesta del sistema
1 Solicita el listado total de usuarios.	2 Lista los usuarios registrados, junto a filtros para posibilitar localizar rápidamente a los usuarios deseados.
3 Selecciona los usuarios que desea actualizar por haber pagado y envía el formulario.	4 Actualiza el estado del pago de los usuarios.
Flujo alternativo	
Acción del actor	Respuesta del sistema
3 Selecciona los usuarios que desea actualizar por haber cancelado el pago y envía el formulario.	3.1 Actualiza el estado del pago de los usuarios.

Caso de uso	
CU - 28	Solicitar nueva contraseña
Propósito	Permitir a los usuarios recuperar las credenciales para acceder al sistema
Actores: Usuario no autenticado	
Resumen: Solicitud para crear una nueva contraseña por el olvido de la actual	
Referencias	R3.3
Escenario 1	
Acción del actor	Respuesta del sistema
1 El usuario accede a la página creada para recuperar contraseña.	2 Se genera el formulario solicitando el nombre de usuario o el correo electrónico de este.
3 Introduce su nombre de usuario o su correo electrónico y envía el formulario.	4 Verifica si existe el nombre de usuario o el correo electrónico en los datos del sistema. 5 Al comprobar la veracidad de los datos envía un

Capitulo II. Caracterización del sistema.

6 Introduce la nueva contraseña en el formulario y envía el mismo.	correo electrónico al usuario identificado con un enlace generado y único mediante el cual se le da acceso al sistema, para que pueda cambiar su contraseña. 7 Actualiza la contraseña del usuario.
Flujo alternativo	
Acción del actor	Respuesta del sistema
	5 Detecta que el usuario o el correo introducido no están almacenados en el sistema y notifica al usuario para que los corrija. Se vuelve al paso 2.

En este capítulo se realizó la modelación de los procesos de negocio, permitiendo lograr la comprensión total del negocio a automatizar, y colaborando a definir los requerimientos que debería cumplir el sistema a desarrollar, así como la descripción de cada uno de los casos de uso que se determinaron.

Capítulo III. Análisis y diseño del sistema.

En el presente capítulo se hace referencia al flujo de trabajo análisis y diseño, donde se refinan los requerimientos y los casos de uso determinados en el capítulo anterior, así como obteniéndose un diseño más específico del software, se definen el modelo de análisis y el diagrama de clases del diseño con los diagramas de interacción correspondientes a cada caso de uso y escenario dentro de estos. Además se incorporan el diagrama de clases persistentes, el modelo de datos, y las descripciones de las tablas correspondientes al modelo de datos.

La metodología propone la realización de un prototipo de interfaz de usuario para lograr un acuerdo entre los equipos de desarrollo y los clientes, que sirva a su vez como parte del contrato entre estos. En la realización de este estudio se elaboraron los respectivos prototipos no funcionales, pero no se encuentran en el presente documento por el volumen que ocuparían.

3.1 Modelo de análisis


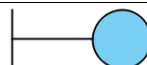
El objetivo del análisis en la metodología RUP es obtener una visión del sistema que se preocupa del “Qué hace”, de modo que solo se interesa en los requisitos funcionales y no como se implementará la solución, en otras palabras, se realiza un refinamiento de los requerimientos y no se toma en cuenta el lenguaje de programación a usar en la construcción.


3.1.1 Diagrama de clases del análisis

Los diagramas de clases del análisis permiten refinar los requisitos capturados, estructurando el sistema en las clases de interfaz, clases de control, y clases de entidad participantes en cada caso de uso, definiendo sus nombres y las relaciones entre ellos.

Estos modelos constituyen una entrada importante para la posterior realización del diseño, dichos diagramas se pueden apreciar en el [anexo 2](#).

Se realizará el modelo de análisis basado en las clases del análisis propuestas por RUP, que se clasifican en:

 Entidad	Clase Entidad: Modelan información que posee larga vida y que a menudo es persistente.
 Interfaz	Clase Interfaz: Modelan la interacción entre el sistema y sus actores.

 Control	Clase control: Coordinan la realización de uno o unos pocos casos de uso coordinando las actividades de los objetos que implementan la funcionalidad del caso de uso.
---	---

3.2 Modelo de diseño

El diseño es un refinamiento del análisis que tiene en cuenta los requisitos no funcionales, en conclusión “Cómo” cumple el sistema su objetivo. Realizar el diseño de forma correcta es suficiente para que el sistema pueda ser desarrollado sin ambigüedades. Cuando la precisión del diseño es amplia, la implementación puede ser hecha por sistemas generadores de código.

3.2.1 Diagrama de clases con estereotipos Web

Los diagramas de clases brindan una visión estática del sistema a desarrollar; describen comportamientos e interacciones entre las clases.

Para el desarrollo de un sistema pueden existir diversos diagramas de clases representando las partes del mismo. Tienen gran importancia ya que permiten mostrar desde diferentes perspectivas la estructura de un sistema visto desde sus clases, y aportan aspectos elementales para la etapa de desarrollo.

Para el desarrollo de aplicaciones Web, RUP propone el uso de nuevos estereotipos, entre los que destacan <<Client Page>> para las páginas clientes o interfaces de usuario, <<Server Page>> para las páginas que se ejecutan en el servidor y <<HTML Form>> para los formularios contenidos en las páginas clientes, entre otros. Este tipo de diagramas, desarrollados para este estudio, pueden ser observados en el [anexo 4](#), donde cada clase controladora está relacionada con el paquete BD, del [anexo 6](#), en el que se encuentran las distintas clases persistentes.

Cada una de las clases utilizadas en, con los atributos y funcionalidades contenidas, son descritas en las tablas a continuación.

Nombre: CI_listadoEventos.page	
Tipo de clase: interfaz	
Atributo	Tipo
idEvento	Int[]
nombreEvento	String[]

Nombre: CI_evento.page

Capitulo III. Análisis y diseño del sistema.

Tipo de clase: interfaz	
Atributo	Tipo
descripcionEvento	String[]
nombreEvento	String[]
nombreCategorias	String[]

Nombre: CI_actividades.page	
Tipo de clase: interfaz	
Atributo	Tipo
idActividad	Int[]
nombreActividad	String[]

Nombre: CI_relatoria.page	
Tipo de clase: interfaz	
Atributo	Tipo
nombreActividad	string
descripcionActividad	string
textoRelatoria	string

Nombre: CI_ponencia.page	
Tipo de clase: interfaz	
Atributo	Tipo
nombreActividad	string
descripcionActividad	string
fechaHora	long

Nombre: CI_galeriaImagenes.page	
Tipo de clase: interfaz	
Atributo	Tipo
nombreActividad	string
descripcionActividad	string
idGaleria	int

Capítulo III. Análisis y diseño del sistema.

Nombre: CC_gestorEvento.php	
Tipo de clase: controladora	
Responsabilidades:	
Nombre:	ObtenerListadoEventos(idEdicion : int)
Descripción:	Obtener a partir del identificador de una edición todos los eventos de esta
Nombre:	ObtenerListadoCategorias(idEvento : int)
Descripción:	Obtener el listado de todas categorías de actividades que existen para un evento determinado, por el atributo identificador del mismo
Nombre:	ObtenerDatosEvento(idEvento : int)
Descripción:	Obtener todos los datos de un evento por su identificador
Nombre:	ObtenerActividades(idCategoriaActividades : int, idEvento : int)
Descripción:	Obtener de un evento y una categoría de actividad todas las actividades existentes

Nombre: CC_gestorActividades	
Tipo de clase: controladora	
Responsabilidades:	
Nombre:	ObtenerDatosActividad(idActividad : int)
Descripción:	Por el identificador de una actividades obtener todos los datos de la misma
Nombre:	ObtenerDatosRelatoria(idActividad : int)
Descripción:	Obtener los datos de una relatoría por su identificador
Nombre:	ObtenerDatosPonencia(idActividad : int)
Descripción:	Obtener los datos de una ponencia por su identificador
Nombre:	ObtenerDatosGaleria(idActividad : int)
Descripción:	Obtener los datos de una galería por su identificador

Nombre: CI_perfilUsuario.page
Tipo de clase: interfaz
Esta clase solo contiene formularios, para configurar los datos de usuario y las preferencias de usuario

Nombre: ajax.js
Tipo de clase: script

Capitulo III. Análisis y diseño del sistema.

Responsabilidades:	
Nombre:	BorrarPreferenciaParticipacion
Descripción:	Mediante un botón colocado al lado de la preferencia, usando ajax, borrar la preferencia de usuario que se seleccione

Nombre: Form_preferenciasUsuario	
Tipo de clase: formulario	
Atributo	Tipo
listadoldPreferencias	Int[]
listadoNombresActividadPreferencia	String[]

Nombre: Form_datosUsuario	
Tipo de clase: formulario	
Atributo	Tipo
contrasenaActual	string
contrasenaNueva	string
contrasenaNuevaRepetida	String
nombres	string
apellidos	string
ocupacion	string
institucion	string
idCiudad	int
calle	string
codigoPostal	string
telefono	string
fax	string
urlPersonal	string
correoElectronico	string
curriculoUrl	string
curriculoNuevo	file

Nombre: CC_gestorPerfilUsuario.php

Capitulo III. Análisis y diseño del sistema.

Tipo de clase: controladora	
Responsabilidades:	
Nombre:	ObtenerDatosUsuario(idUsuario : int)
Descripción:	Obtener los datos de un usuarios por su atributo identificador
Nombre:	ObtenerPreferenciasParticipacion(idUsuario : int)
Descripción:	Obtener las actividades de preferencia de un usuario por su identificador
Nombre:	ActualizarDatosUsuario(datosPersonalesUsuaio : datosUsuario)
Descripción:	Actualizar los datos del usuario
Nombre:	BorrarPreferenciaParticipacion(idUsuario : int, idPreferencia : int)
Descripción:	Borrar una preferencia dado el identificador del usuario y de la preferencia que se desea borrar

Nombre: CI_enviarTrabajo.page	
Tipo de clase: interfaz	
Atributo	Tipo
listadoTitulo	String[]
listadoResumen	String[]
listadoPalabrasClaves	String[]
listadoldFichero	String[]
listadoModalidadPropuesta	String[]
listadoComisionPropuesta	String[]
listadoExplicacionEstado	String[]

Nombre: Form_envioTrabajo	
Tipo de clase: formulario	
Atributo	Tipo
titulo	string
resumen	string
palabrasClaves	string
fichero	file
idModalidadPropuesta	int
idComisionPropuesta	int

Capítulo III. Análisis y diseño del sistema.

Nombre: CC_gestorTrabajosUsuario.php	
Tipo de clase: controladora	
Responsabilidades:	
Nombre:	ObtenerTrabajosEnviados(idUsuario : int)
Descripción:	Obtener todos los trabajos enviados por el usuario, con el estado en que se encuentre cada cual
Nombre:	AlmacenarTrabajoCandidato(datosNuevoTrabajo : datosTrabajo)
Descripción:	Guardar un trabajo candidato a la convención
Nombre:	ActualizarEstadoTrabajo(idTrabajo : int, idNuevoEstado : int, explicacionCambioEstado : string)
Descripción:	Para actualizar el estado de un trabajo cuando el comité científico lo analice

Nombre: CI_listadoEdiciones.page	
Tipo de clase: interfaz	
Atributo	Tipo
listadoNombreEdicion	String[]
listadoIdEdicion	Int[]
listadoAnno	Int[]

Nombre: CI_editarEdicion.page	
Tipo de clase: interfaz	
Esta clase solo contiene el formulario de editar los datos de las ediciones	

Nombre: Form_editarEdicion	
Tipo de clase: formulario	
Atributo	Tipo
idEdicion	int
nombreEdicion	string
descripcionEdicion	string
disponible	boolean
porDefecto	boolean

Capitulo III. Análisis y diseño del sistema.

Nombre: CI_crearEdicion.page
Tipo de clase: interfaz
Esta clase solo contiene el formulario de crear las ediciones

Nombre: Form_crearEdicion	
Tipo de clase: formulario	
Atributo	Tipo
nombreEdicion	string
descripcionEdicion	string
disponible	boolean
porDefecto	boolean

Nombre: Form_borrarEdicion	
Tipo de clase: formulario	
Atributo	Tipo
idEdicionBorrar	Int[]

Nombre: CC_gestorEdiciones.php	
Tipo de clase: controladora	
Responsabilidades:	
Nombre:	ObtenerListadoEdiciones()
Descripción:	Obtener el listado de las ediciones existentes en el sistema
Nombre:	EditarEdicion(datosEdicion : edicion)
Descripción:	Editar los datos de una edición almacenada
Nombre:	CrearEdicion(datosEdicion : edicion)
Descripción:	Crear una edición
Nombre:	BorrarEdiciones(idEdiciones : int[])
Descripción:	Borrar alguna de las ediciones existentes

Nombre: CI_listadoSolicitudesStand.page
Tipo de clase: interfaz

Capitulo III. Análisis y diseño del sistema.

Atributo	Tipo
listadoNombresUsuarioConSolicitud	String[]
listadoIdSolicitud	Int[]

Nombre: CI_procesarSolicitudStand.page	
Tipo de clase: interfaz	
Atributo	Tipo
nombreUsuario	string
curriculoUsuario	string

Nombre: Form_cambioEstadoSolicitud	
Tipo de clase: formulario	
Atributo	Tipo
idSolicitud	int
idNuevoEstado	int
explicacionCambioEstado	string

Nombre: CC_gestorSolicitudStand.php	
Tipo de clase: controladora	
Responsabilidades:	
Nombre:	ObtenerSolicitudesRealizadas()
Descripción:	Obtener las solicitudes de stand realizadas
Nombre:	ObtenerDatosSolicitud(idSolicitud)
Descripción:	Obtener por el identificador de una solicitud los datos de la misma
Nombre:	ActualizarEstadoSolicitud(idSolicitud : int, idNuevoEstado : int, explicacionCambioEstado : string)
Descripción:	Actualizar el estado de una solicitud de stand cuando el organizador de la feria le cambia el estado a la misma
Nombre:	CrearSolicitudStand(detallesNuevaSolicitud : string, idUsuarioSolicitud : int)
Descripción:	Crear una solicitud de stand para la feria

Nombre: CI_listarTranajosCandidatos.page

Capitulo III. Análisis y diseño del sistema.

Tipo de clase: interfaz	
Atributo	Tipo
listadoEnviadoIdTrabajo	Int[]
listadoEnviadoTitulo	String[]
listadoEnviadoNombreUsuario	String[]
listadoEnviadoNombreEstado	String[]

Nombre: CI_procesarTrabajoCandidato.page	
Tipo de clase: interfaz	
Atributo	Tipo
idTrabajo	int
titulo	string
resumen	string
palabrasClaves	string
linkFichero	string
nombreUsuario	string
linkCurriculo	string
nombreModalidad	string

Nombre: Form_estadoTrabajo	
Tipo de clase: formulario	
Atributo	Tipo
idTrabajo	Int
idNuevoEstado	int
explicacionNuevoEstado	string
idModalidadPresentacion	int

Nombre: CI_solicitarStand.page	
Tipo de clase: interfaz	
Atributo	Tipo
listadoEnviadoDetalles	String[]
listadoEnviadoNombreEstado	String[]

Capítulo III. Análisis y diseño del sistema.

listadoEnviadoExplicacionEstado	String[]
---------------------------------	----------

Nombre: javascript.js	
Tipo de clase: script	
Responsabilidades:	
Nombre:	ponerDetallesStandSeleccionado()
Descripción:	Al seleccionar en el mapa una región por este método se coloca la información del mismo en el campo de descripción de la solicitud

Nombre: Form_solicitudStand	
Tipo de clase: formulario	
Atributo	Tipo
detallesNuevaSolicitud	String

3.2.2 Diagramas de interacción

Los diagramas de interacción se utilizan para modelar aspectos dinámicos de un sistema, mostrando la realización de un flujo particular de los casos de uso, con la modelación de clases y los mensajes enviados entre ellas, conteniendo de esta forma detalles de la implementación. Para este estudio se realizaron como diagrama de interacción, los diagramas de secuencia de los casos de uso arquitectónicamente significativos, los cuales se encuentran en el [anexo 3](#).

3.2.3 Diseño de la BD

El Modelo Entidad Relación juega un papel importante en el desarrollo de software ya que “permite desarrollar un diseño de base de datos en un esquema de alto nivel conceptual sin considerar los problemas de bajo nivel como la eficiencia, el modelo implícito del administrador de base de datos o las estructuras físicas de los datos” (SAIEDIAN). Se puede apreciar el modelo entidad relación en el [anexo 5](#), el cual se dividió en dos para una mejor visión en este documento.

En este capítulo se realizó, siguiendo la metodología RUP, el modelo de análisis del sistema, donde se modelaron las interfaces, clases controladoras y entidades del análisis, que fueron refinadas en el diseño mediante los diagramas de clases del diseño con estereotipos Web, y posteriormente en los

Capítulo III. Análisis y diseño del sistema.

diagramas de interacción, donde se modeló el dinamismo del sistema; diagramas que permitieron determinar las clases persistentes que posteriormente se transformaron en el modelo entidad relación.

Conclusiones.

Se realizó un estudio profundo y crítico del estado del arte y otros temas que forman parte de la fundamentación teórica, labor que permitió definir la metodología, tecnologías, herramientas y lenguajes más propicios para el desarrollo del producto. Al culminar esta actividad se modelaron los procesos que tenían lugar en el negocios a automatizar, para permitir una comprensión total del mismo y estar en condiciones de pasar a realizar el análisis y diseño del sistema, basados en los requerimientos deseados por los clientes.

Al culminar el presente estudio se puede constatar que se dio cumplimiento al objetivo general de elaborar el análisis y diseño del sistema Web para Informática 2009, y se confirmó la idea de que dicha propuesta permitiría a los desarrolladores implementar el sistema sin contratiempos, ya que el producto fue desarrollado con éxito.

Recomendaciones.

Se recomienda para futuras versiones: automatizar el control del pago de los usuarios, de forma tal que la plataforma de pago se integre al sistema y se tenga sincronizado el estado del pago sin intervención de un usuario cumpliendo tal tarea; realizar un mayor control auditor sobre todas las tareas que realizan en el sistema los distintos usuarios, específicamente de tareas como la aprobación o desaprobación de trabajos, recintos feriales, preguntas y opiniones, e informaciones sobre el bloqueo y desbloqueo de los usuarios; realizar módulo de estadísticas con mayor nivel de detalles que permita desarrollar estudios de mercado u otras acciones encaminadas al perfeccionamiento del sistema; y realizar pruebas de calidad al producto final.

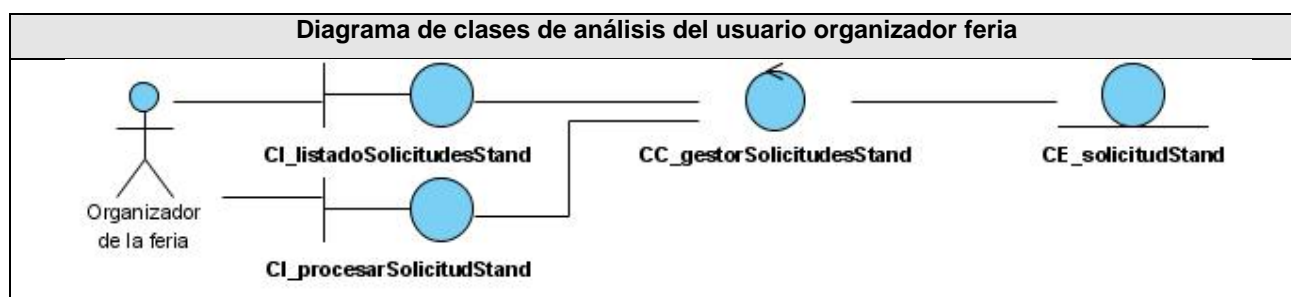
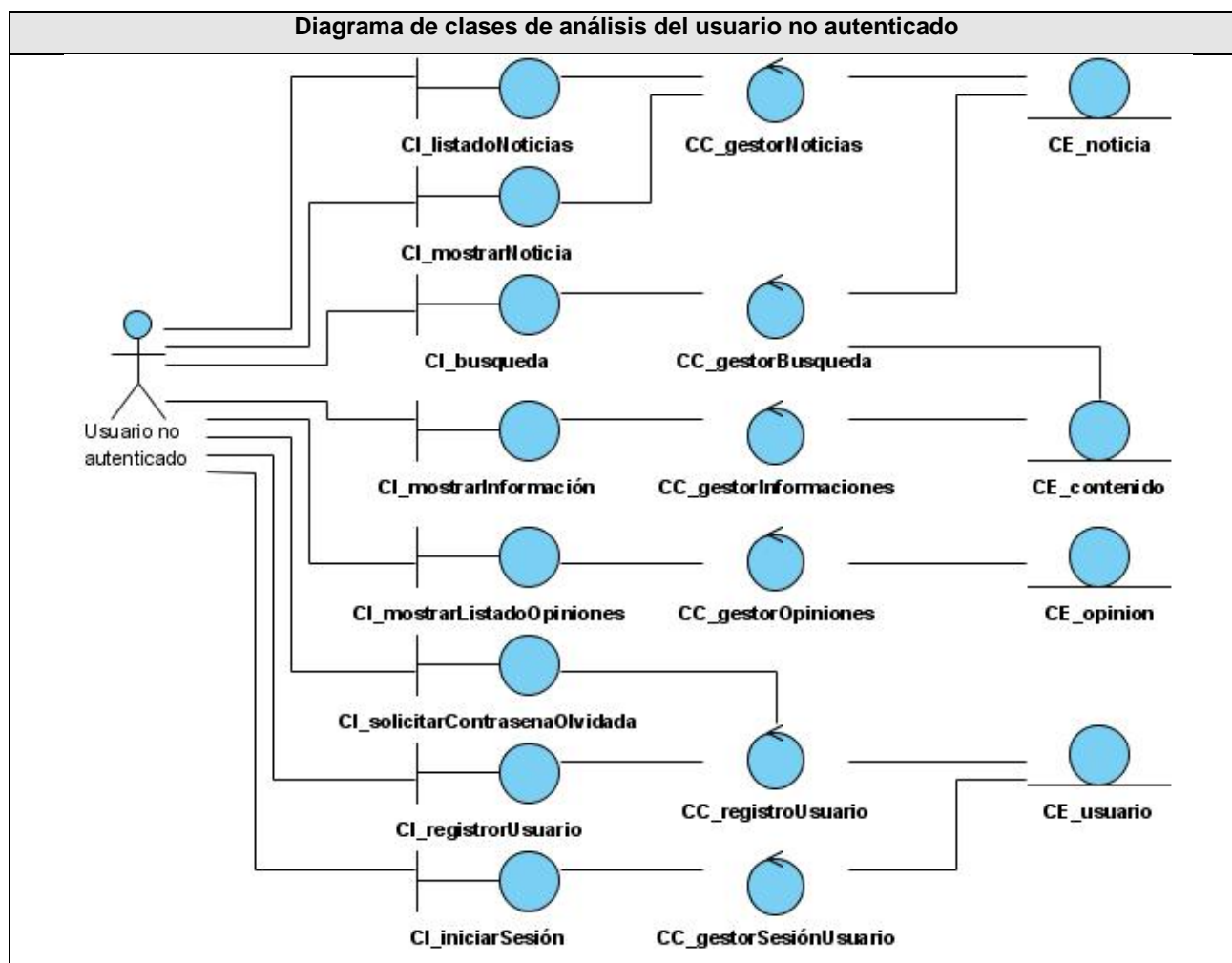
Anexos

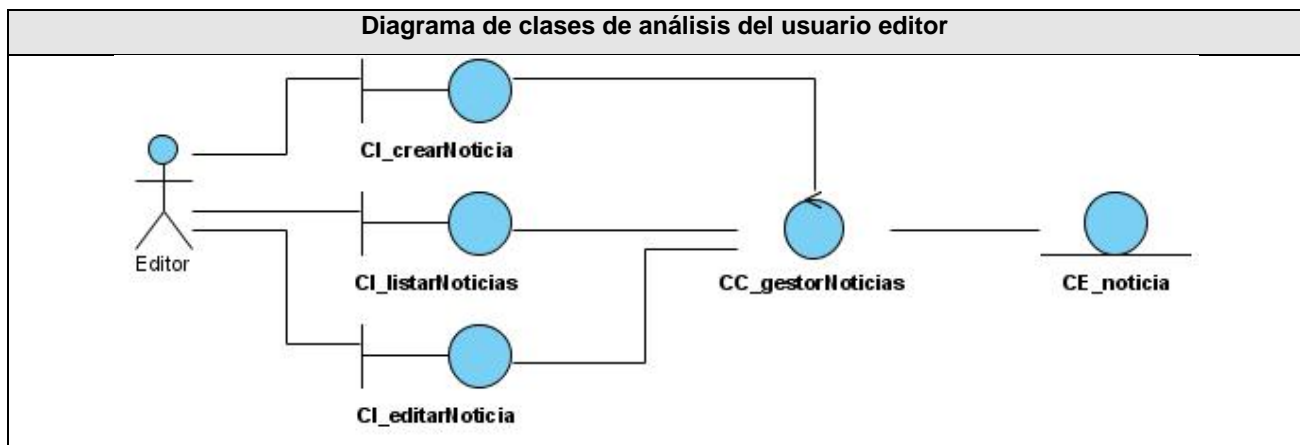
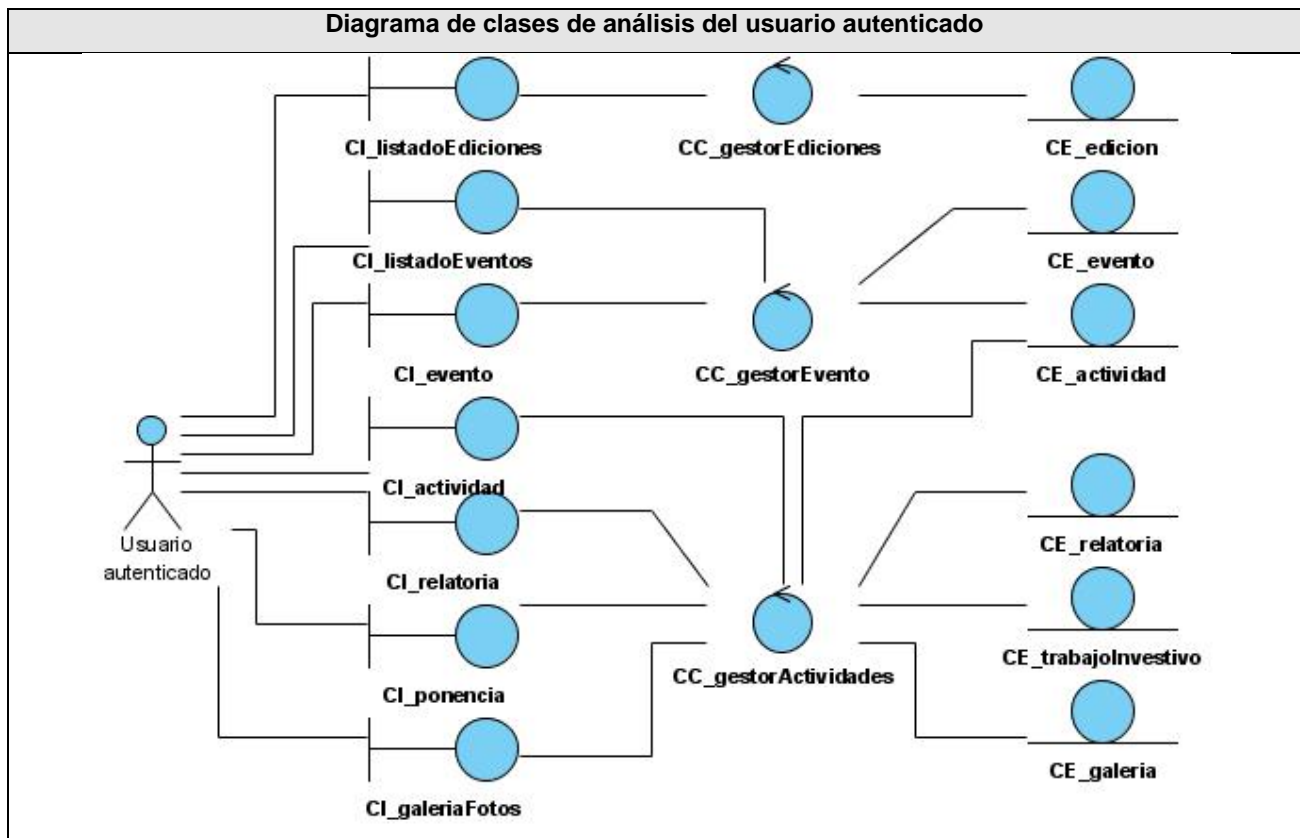
Anexo 1. Tabla comparativa entre CMS.

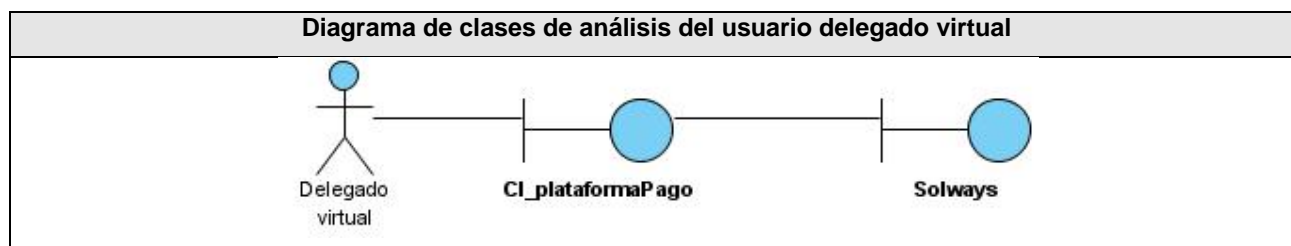
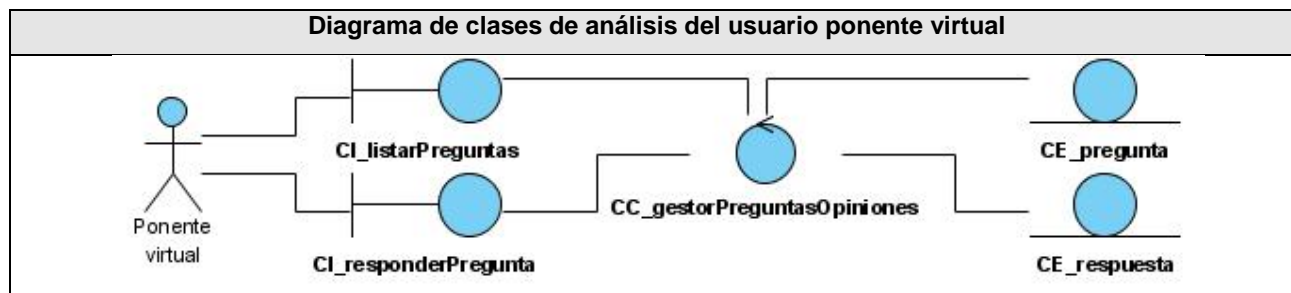
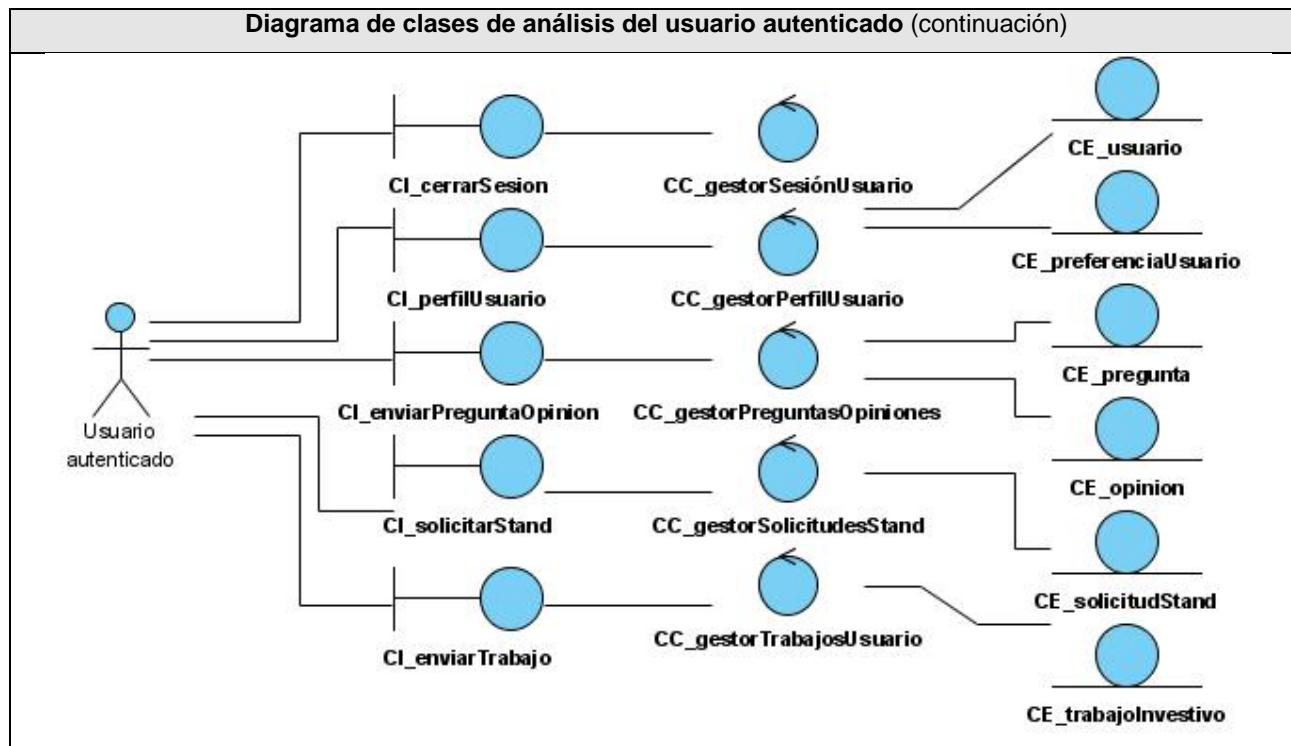
Requerimientos de sistema	CMS Made Simple 1.0.4	Drupal 5.7	E107 0.7.5	Joomla 1.0.7	Mambo 4.6.1	PHP-Fusion 6.00.206
Costo	libre	libre	libre	Libre	Libre	libre
Bases de datos que soporta	MySQL, PostgreSQL	MySQL, PostgreSQL	MySQL	MySQL	MySQL	MySQL
Licencia	GNU GPL + opcional Licencia Comercial	GNU GPL	GNU GPL	GNU GPL	GNU GPL	GNU GPL
Sistema operativo	Multiplataforma	Multiplataforma	Multiplataforma	Multiplataforma	Multiplataforma	Multiplataforma
Servidor de aplicaciones	Cualquier servidor de PHP	Apache, Internet Information Server(IIS)	Apache, Internet Information Server(IIS)	Apache	Cualquier servidor de PHP	Cualquier servidor de PHP
Seguridad	CMS Made Simple 1.0.4	Drupal 5.7	E107 0.7.5	Joomla 1.0.7	Mambo 4.6.1	PHP-Fusion 6.00.206
Registro de acciones	Si	Si	Si	No	No	No
Confirmación de correo	No	Si	Si	Si	Si	Si
Autenticación LDAP	No	Usando componente agregado	Si	No	Si	No
Soporte	CMS Made Simple 1.0.4	Drupal 5.7	E107 0.7.5	Joomla 1.0.7	Mambo 4.6.1	PHP-Fusion 6.00.206

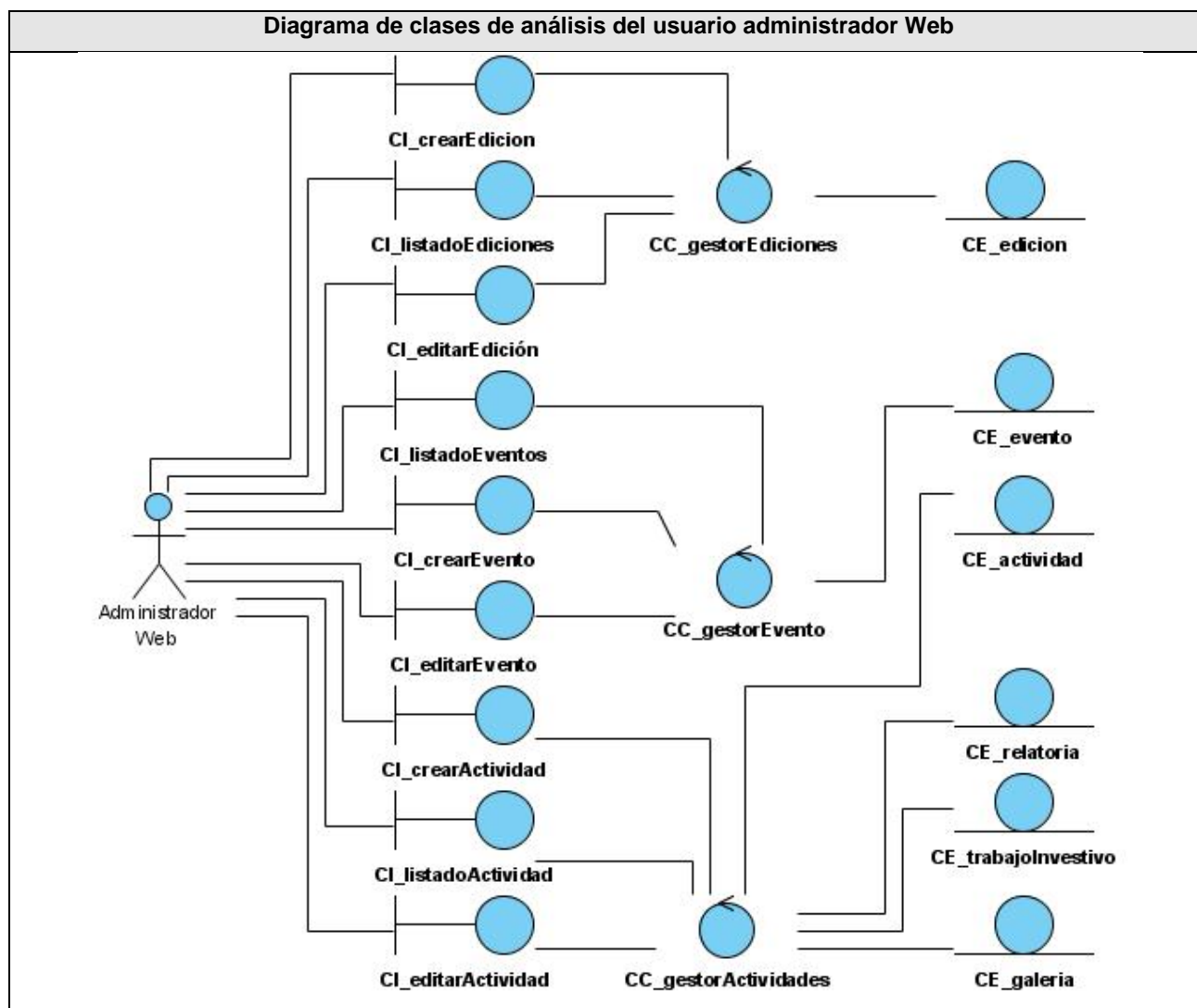
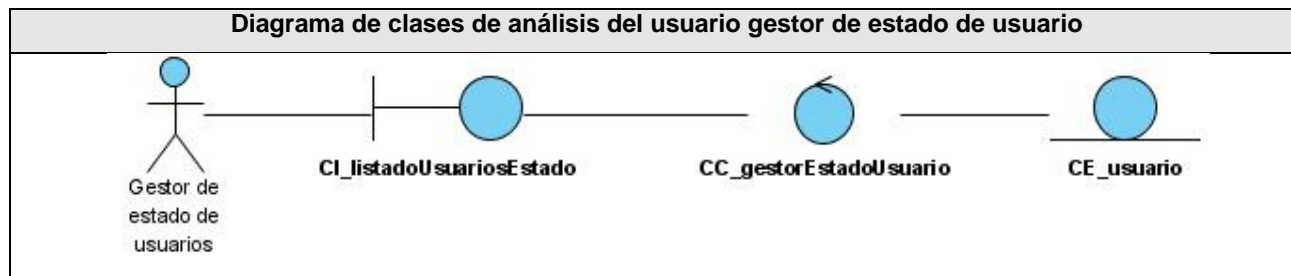
Soporte comercial	No	Si	No	Si	Si	Si
Comunidad de desarrolladores	Si	Si	Si	Si	Si	Si
Ayuda en línea	Si	Si	Si	Si	Si	Limitado
Foros públicos	Si	Si	Si	Si	Si	Si
Funcionamiento	CMS Made Simple 1.0.4	Drupal 5.7	E107 0.7.5	Joomla 1.0.7	Mambo 4.6.1	PHP-Fusion 6.00.206
Balance de carga	No	Si	Si	No	No	No
Administración	CMS Made Simple 1.0.4	Drupal 5.7	E107 0.7.5	Joomla 1.0.7	Mambo 4.6.1	PHP-Fusion 6.00.206
Análisis estadístico	Usando componente agregado	Si	Si	Si	Si	No

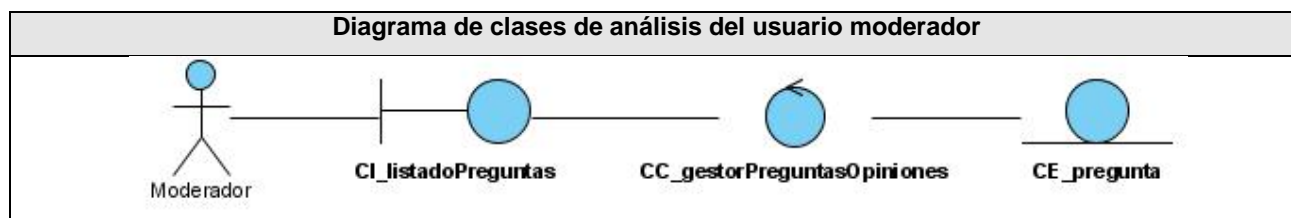
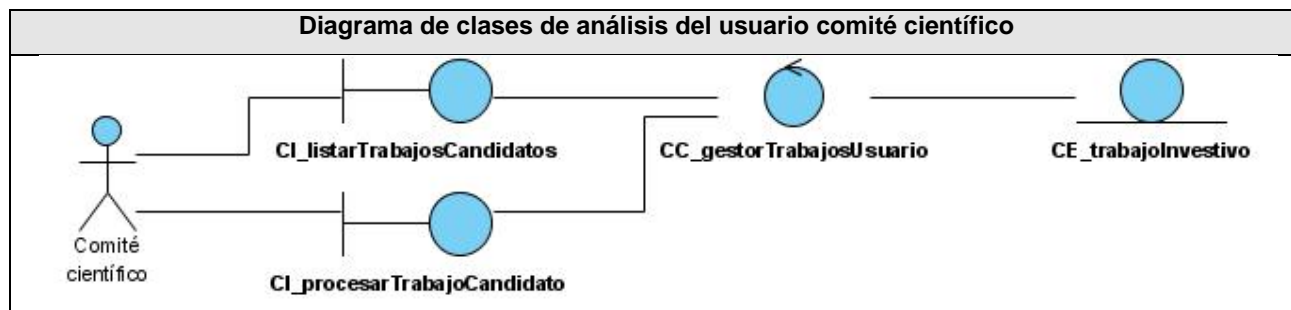
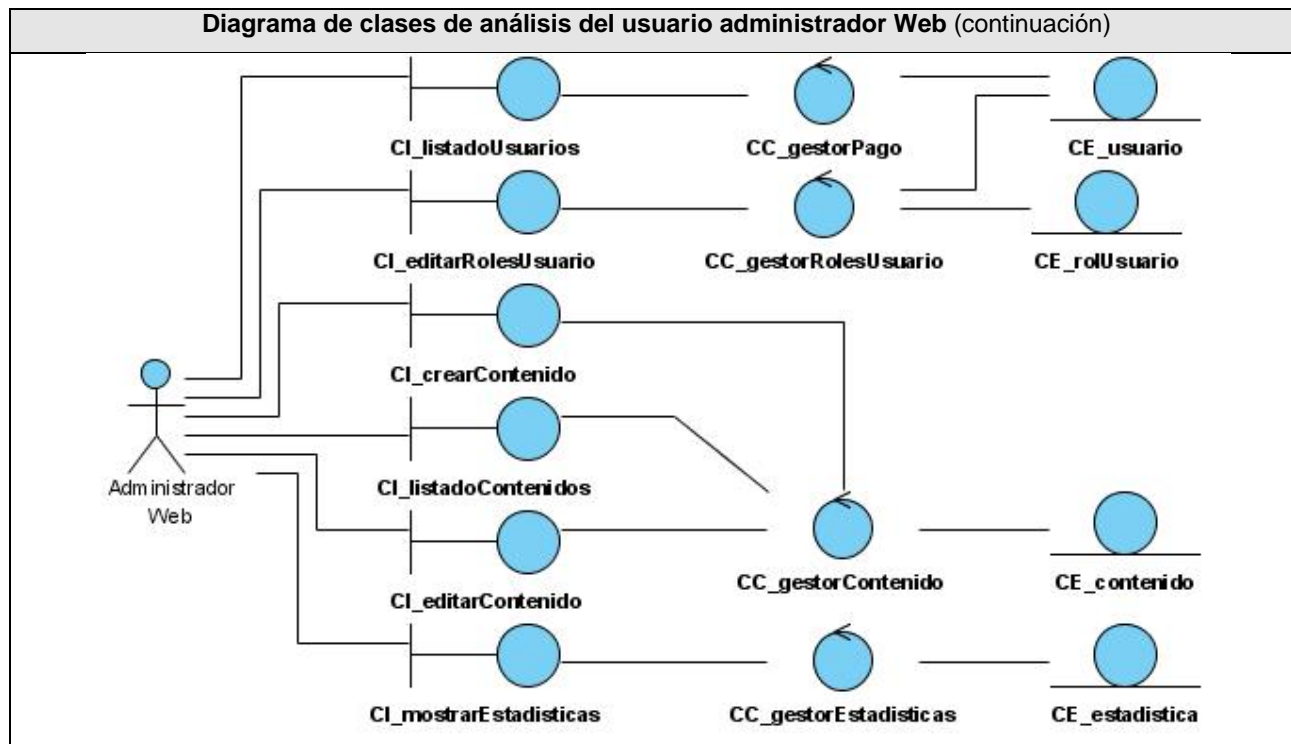
Anexo2. Diagramas de clases.



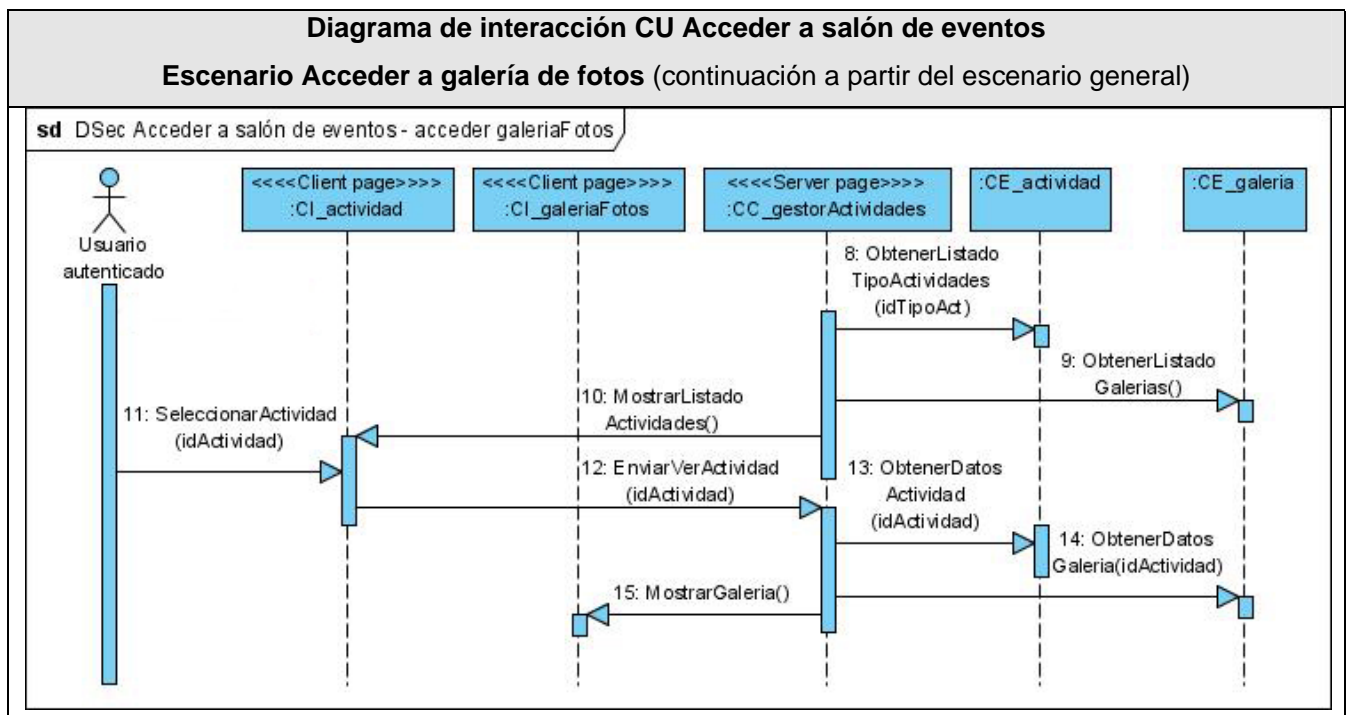
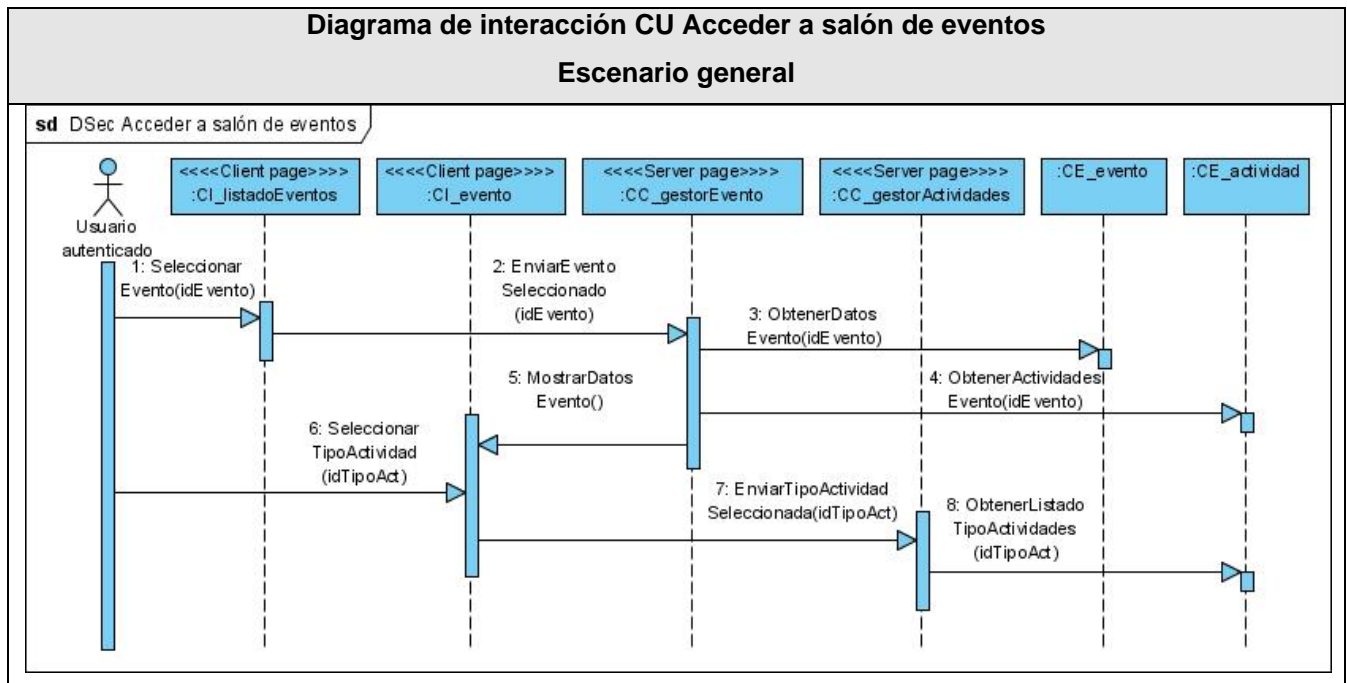


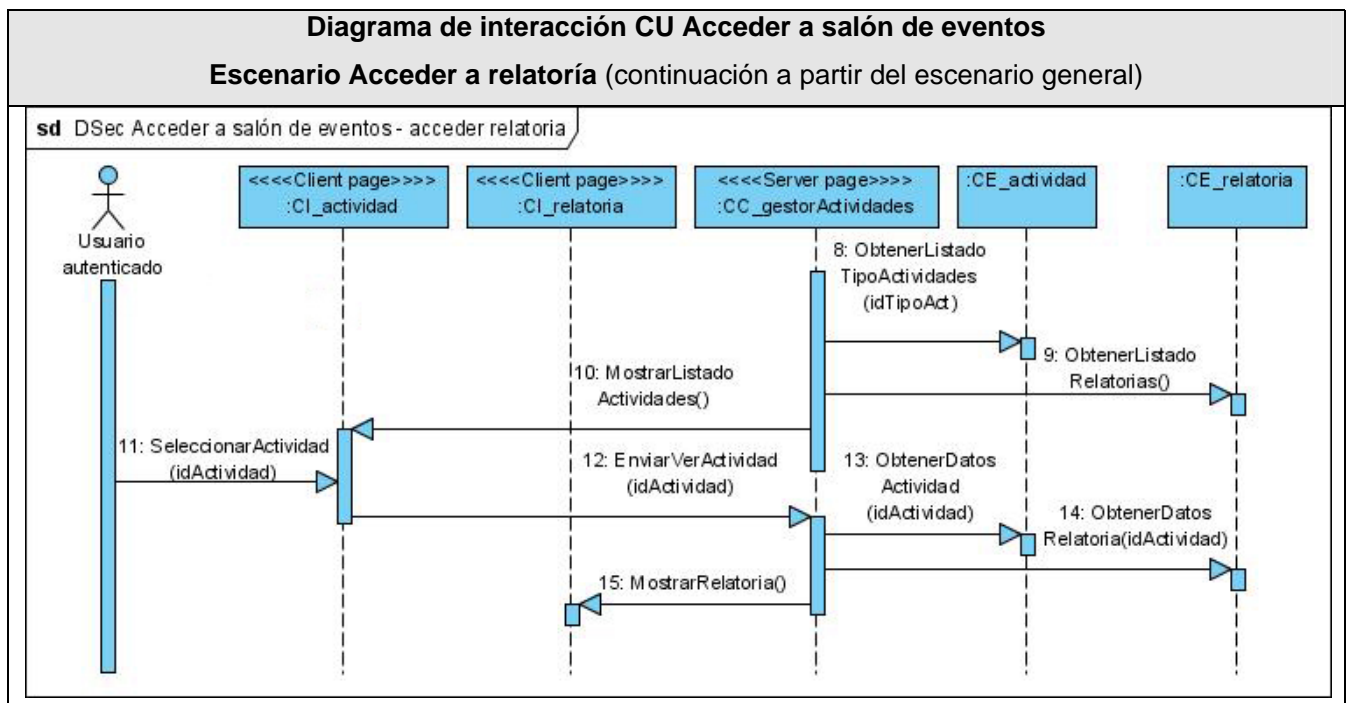
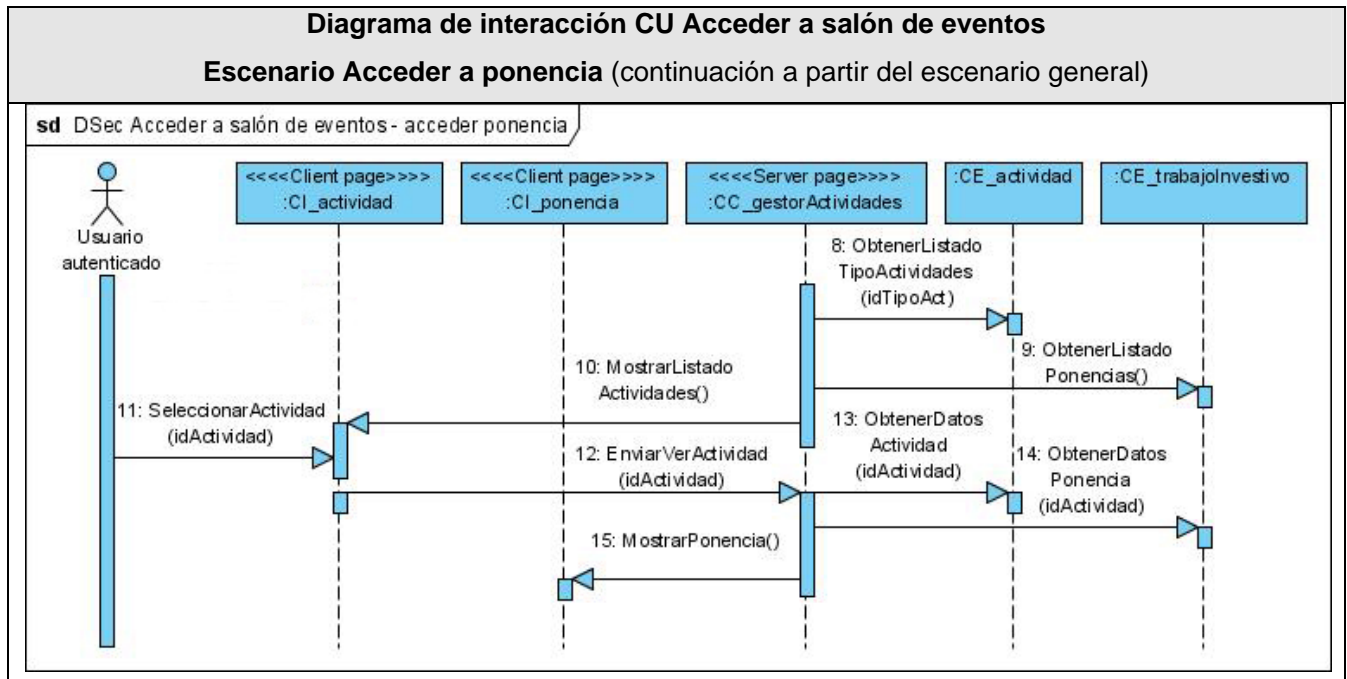


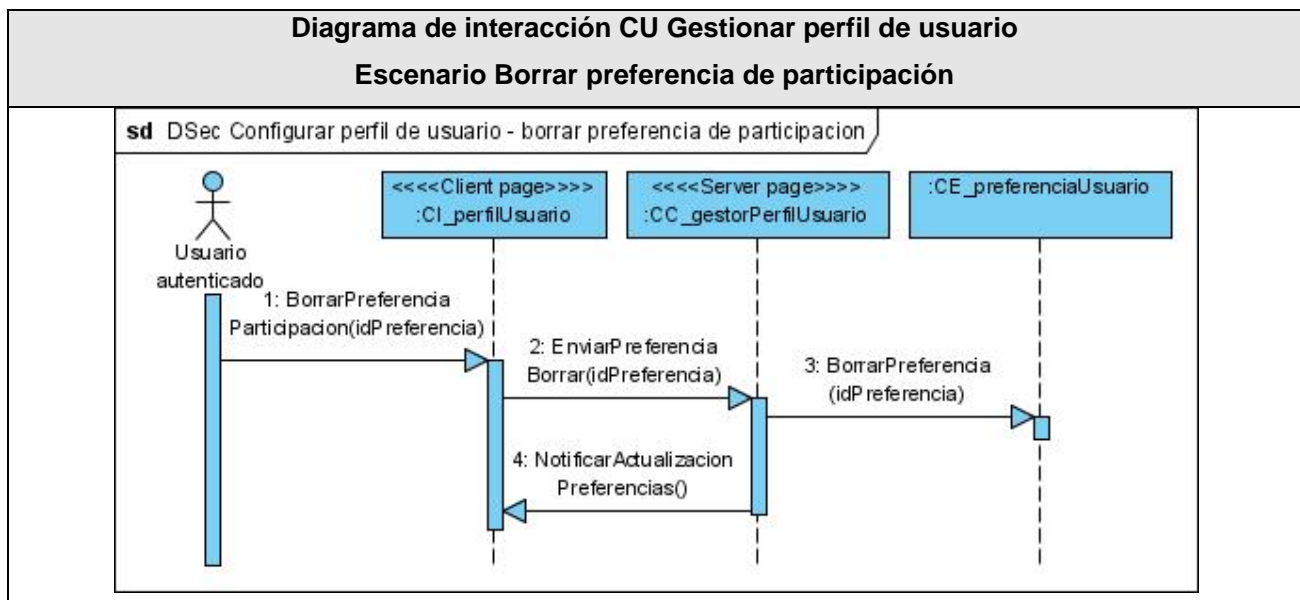
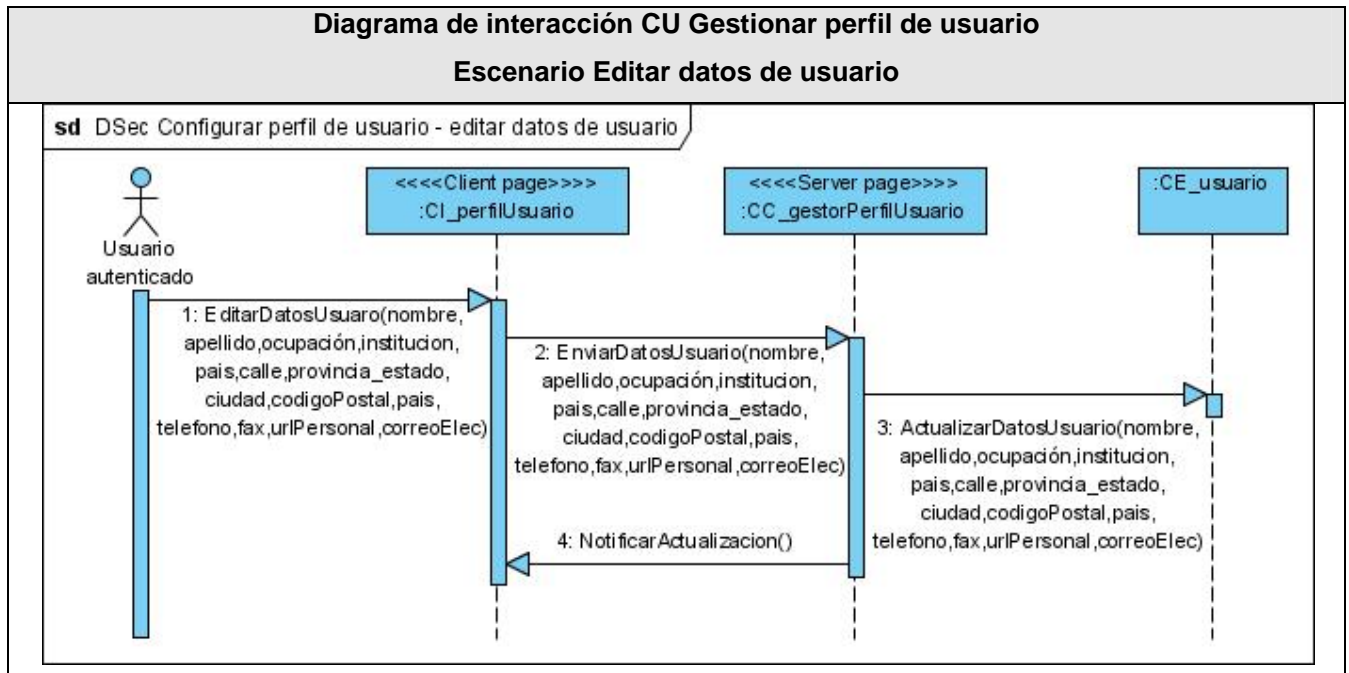


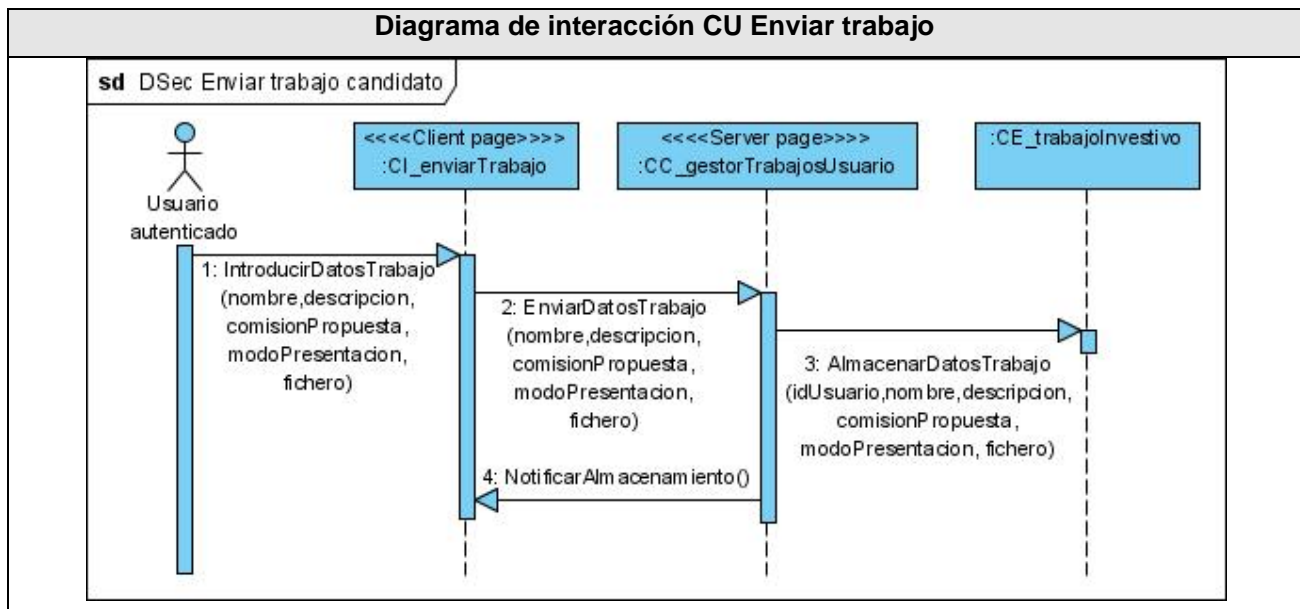
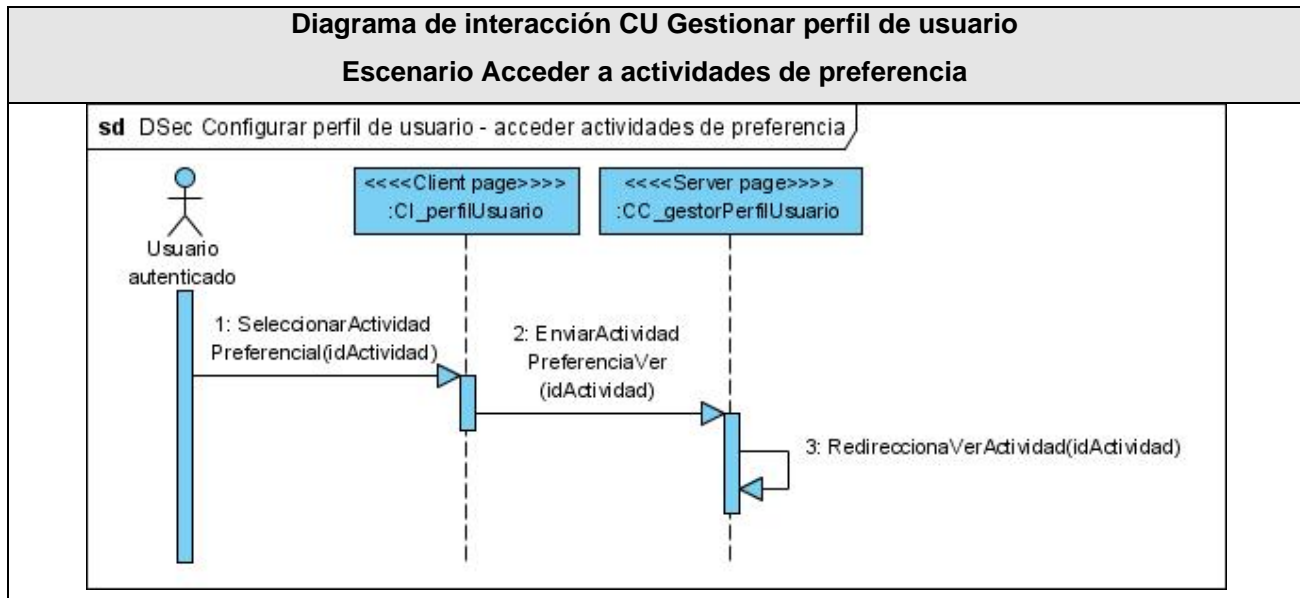


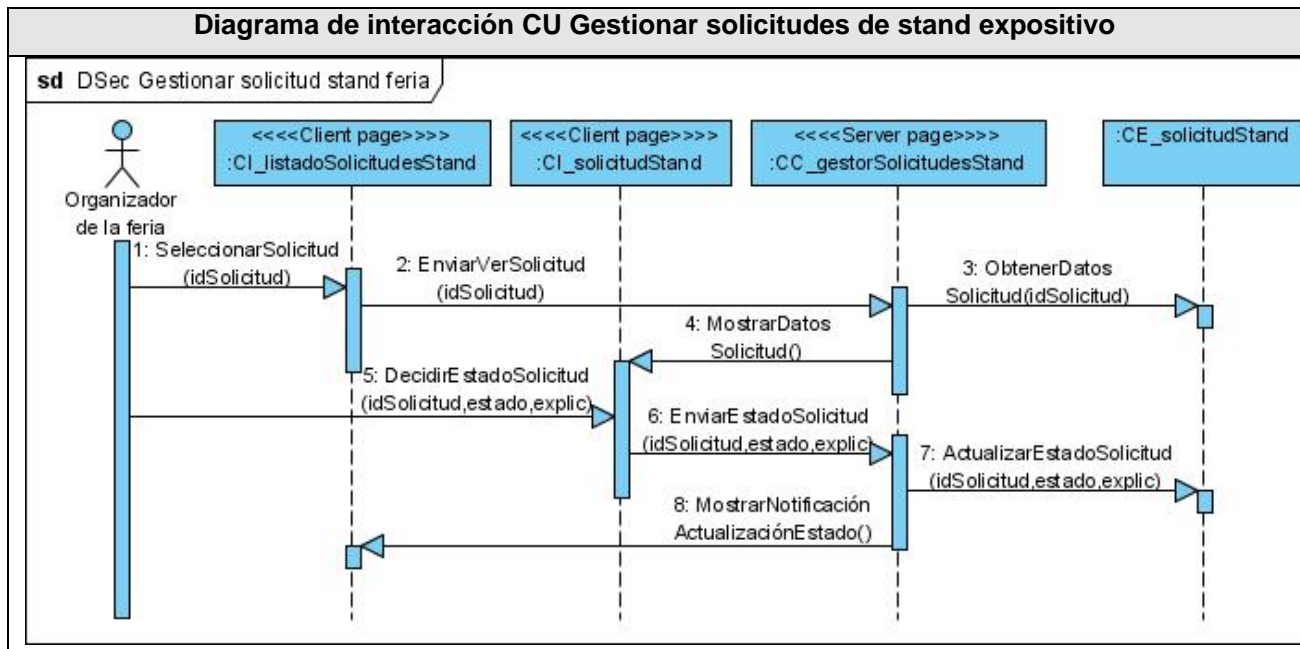
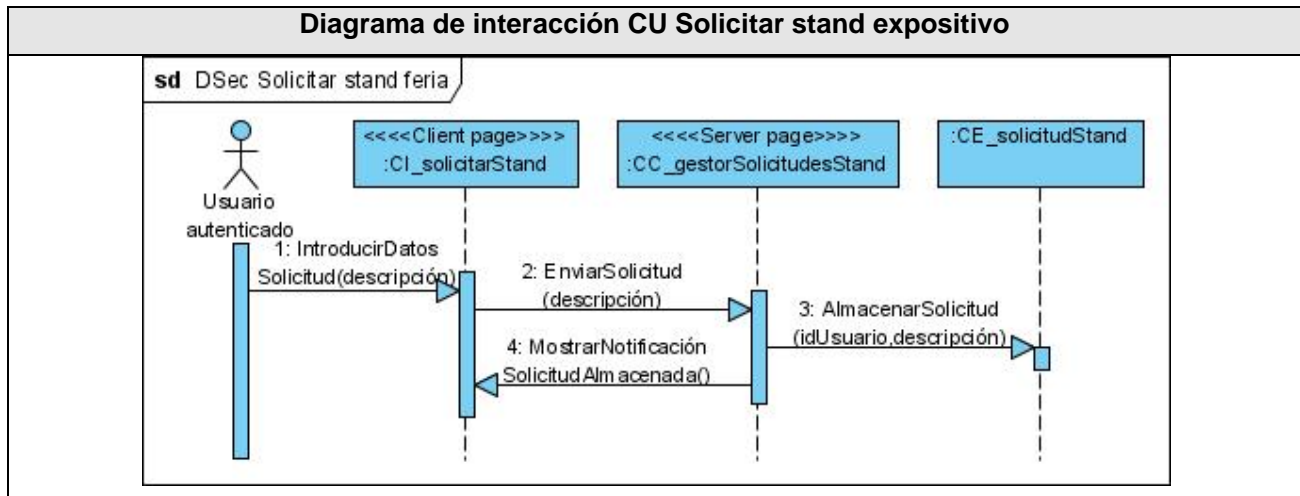
Anexo3. Diagramas de interacción.

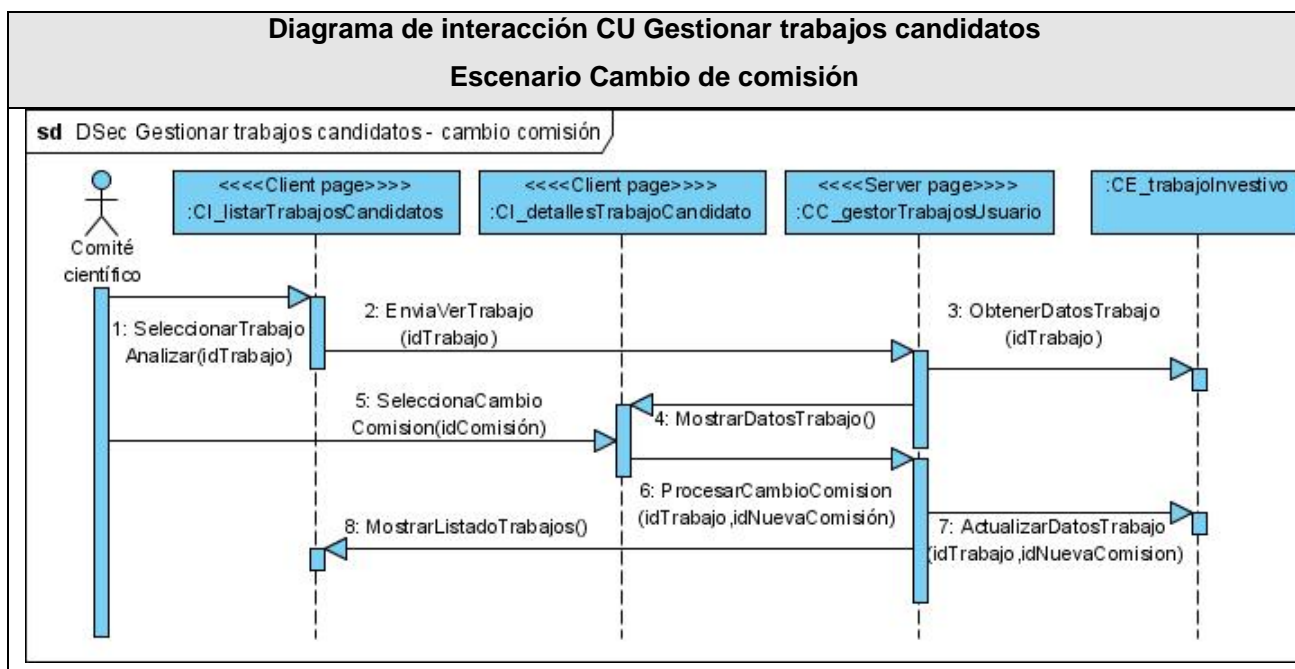
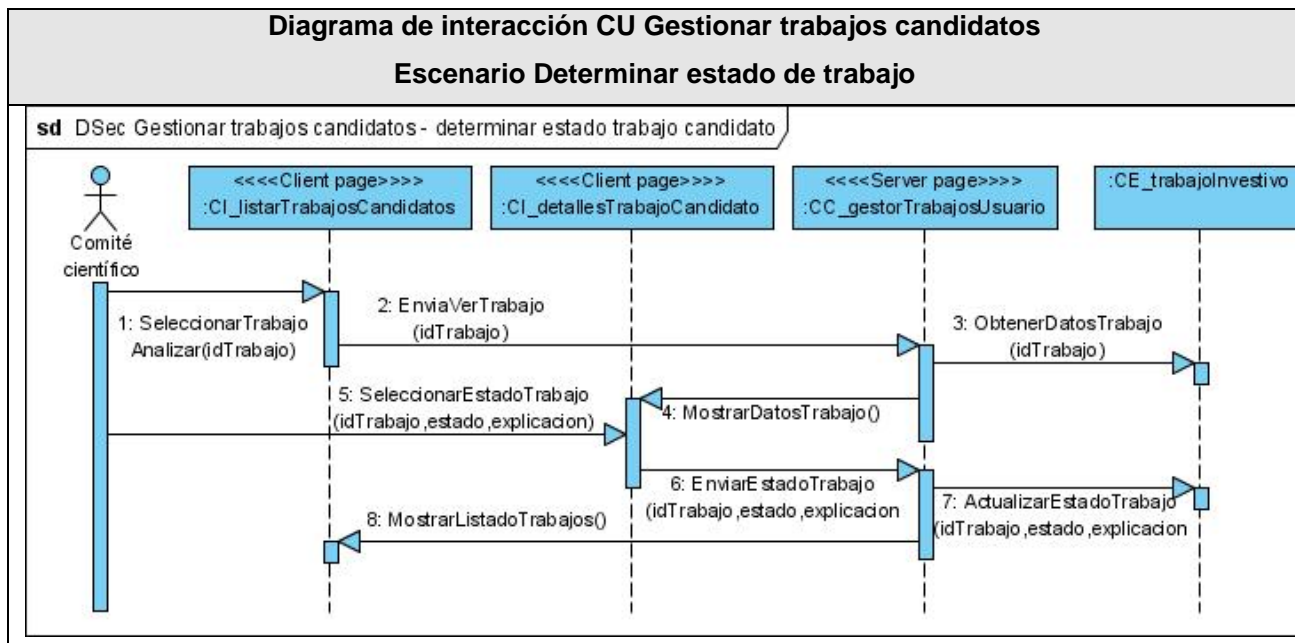


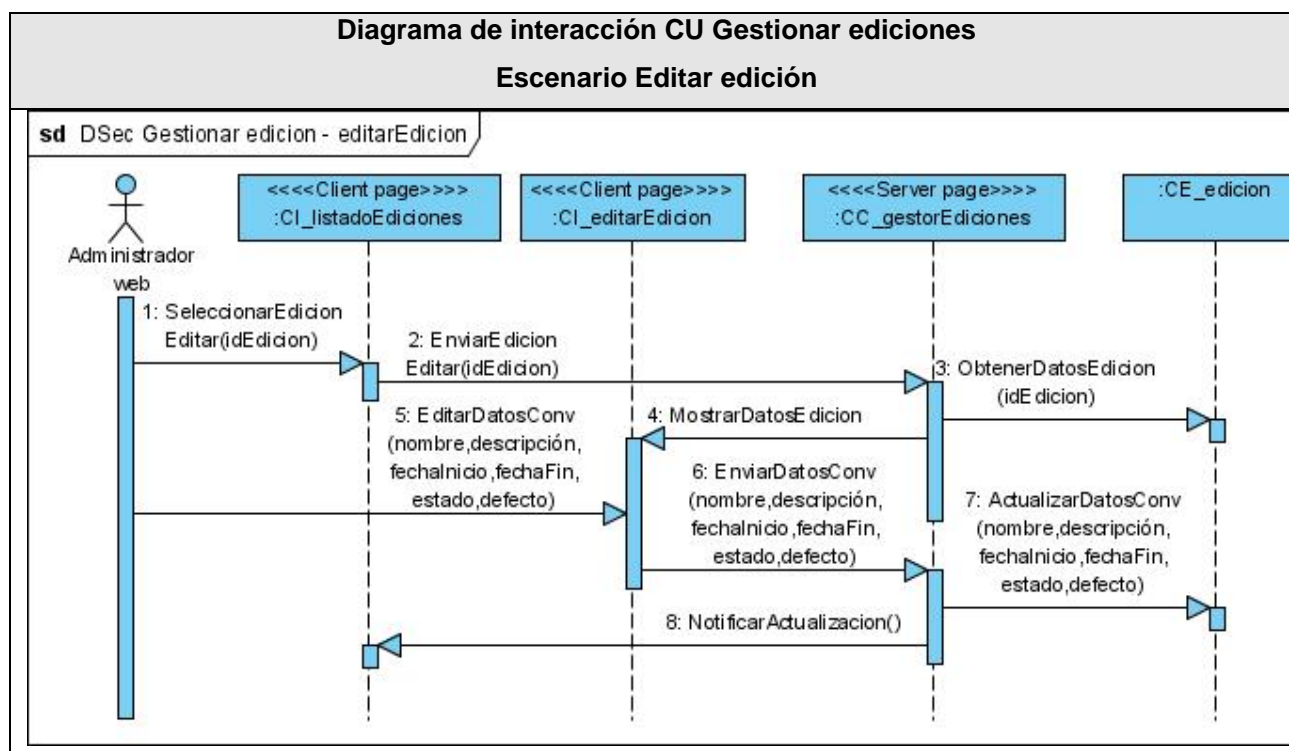
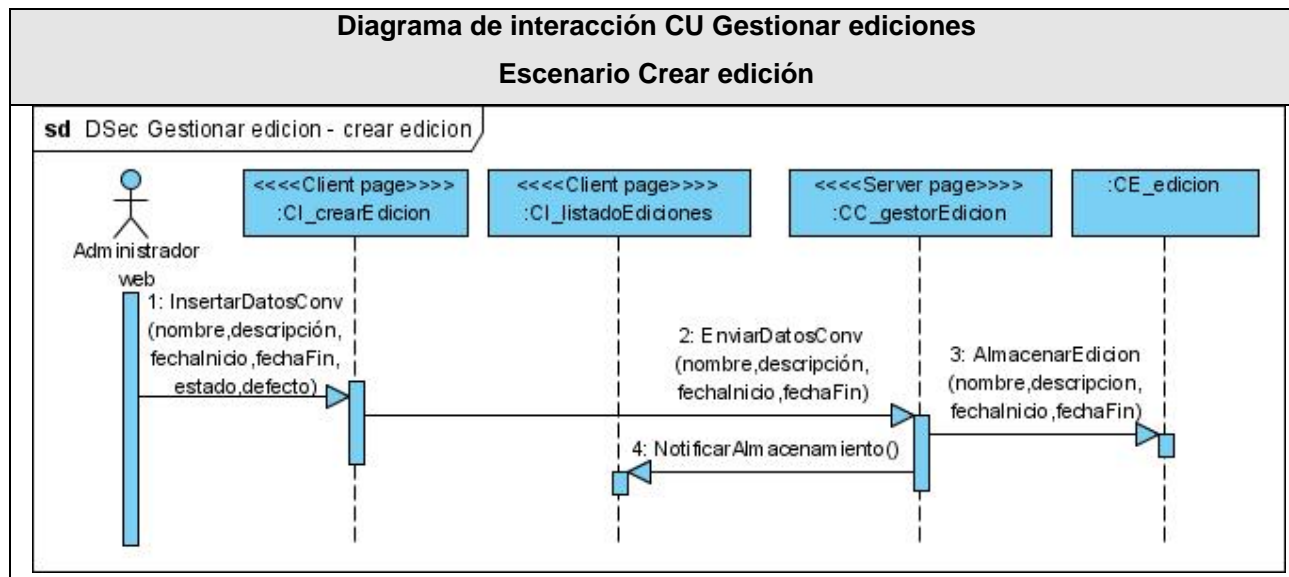


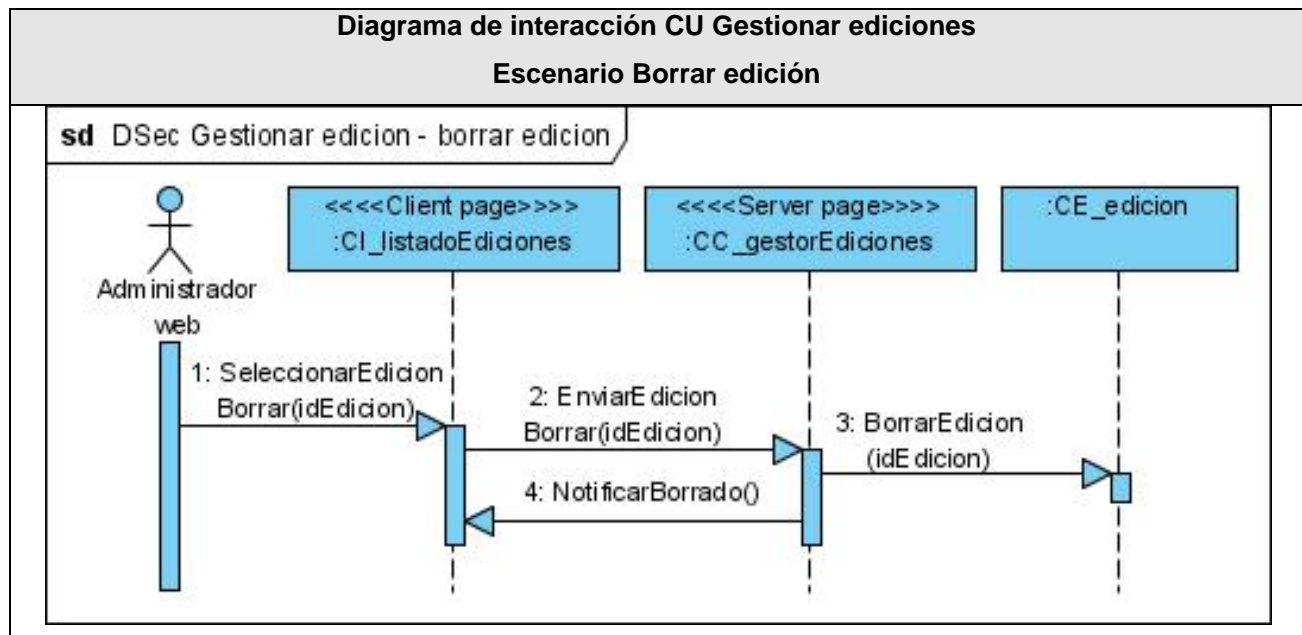




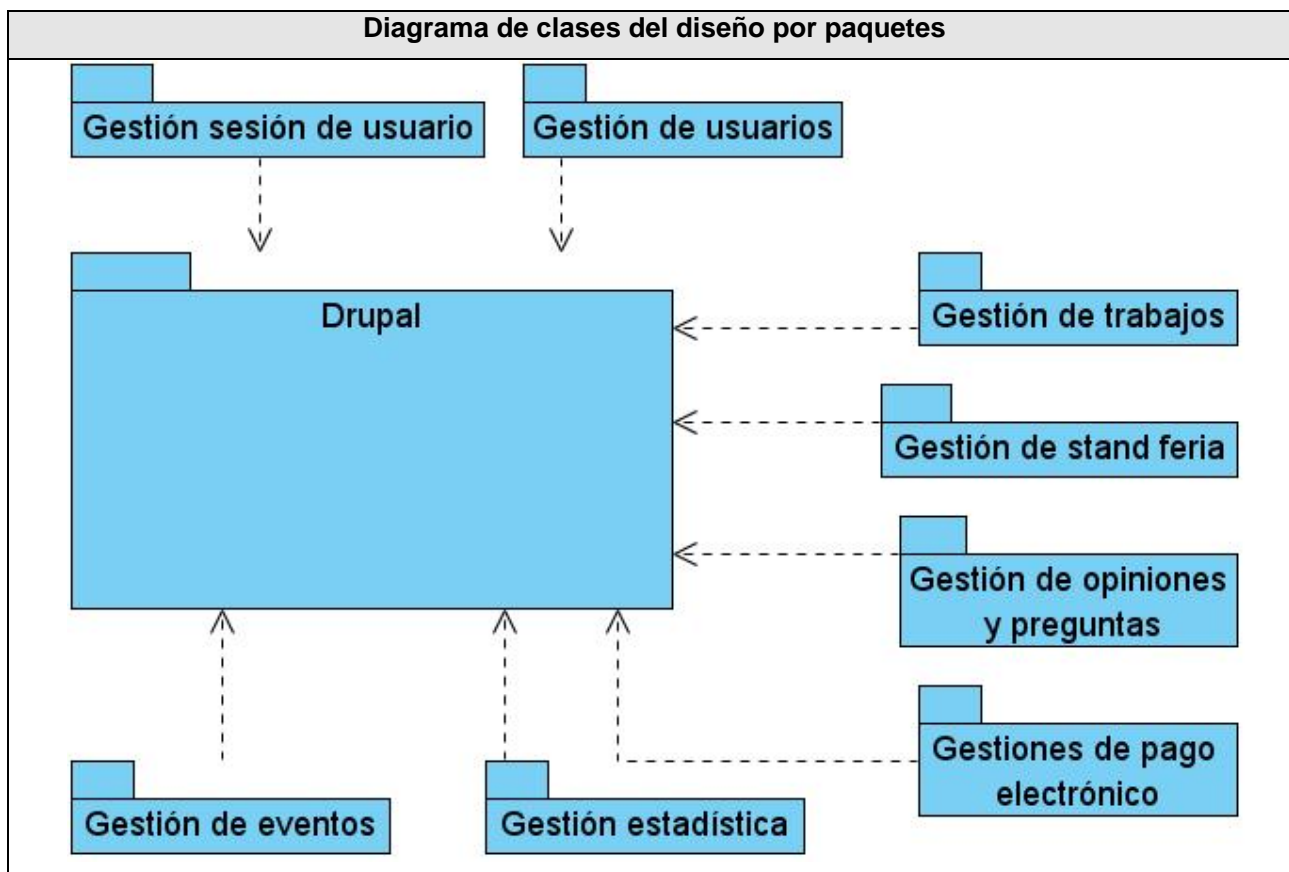


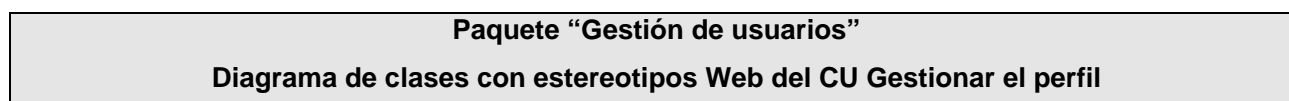
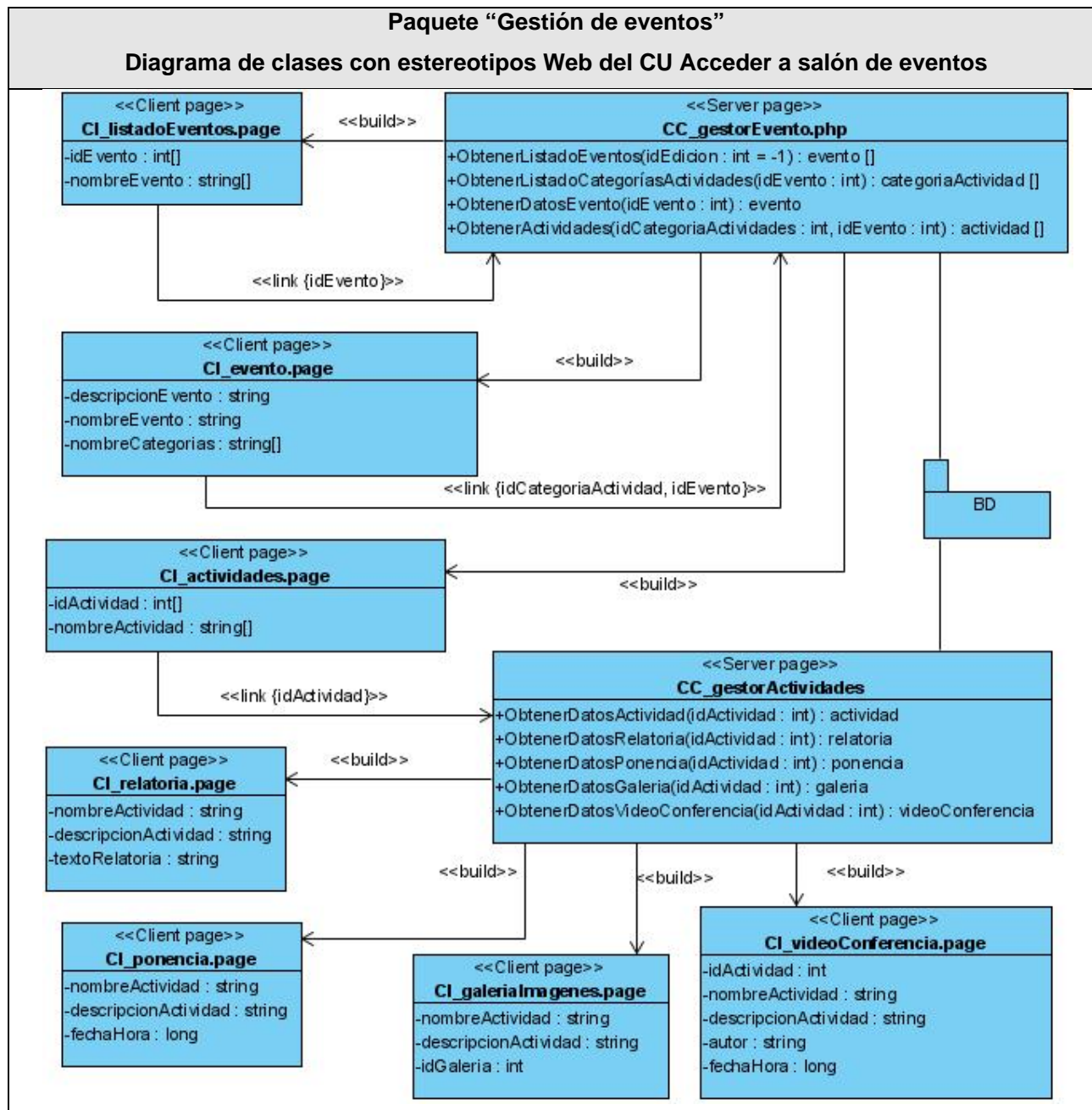


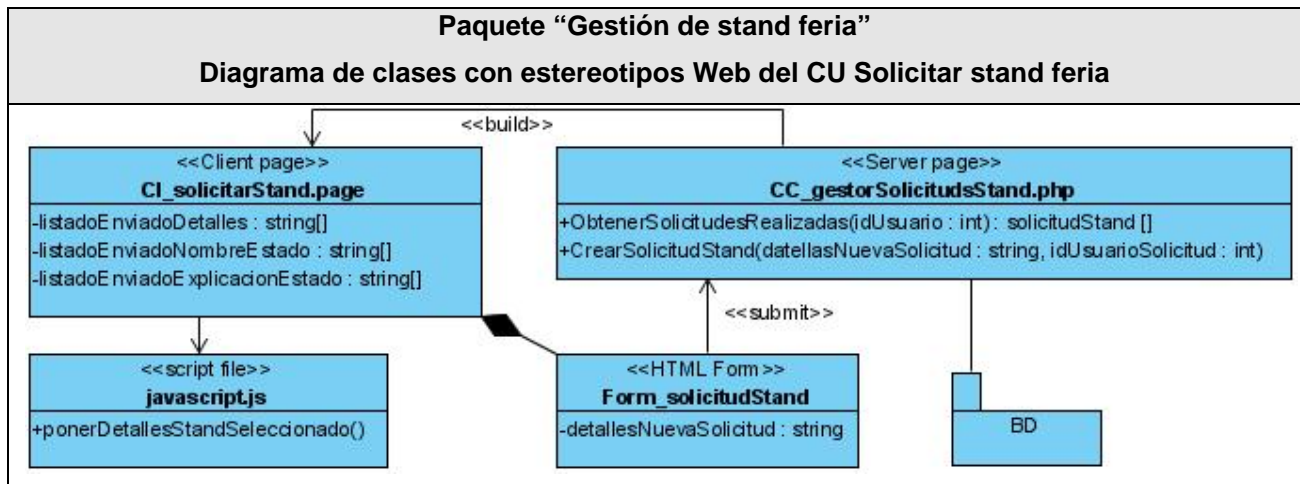
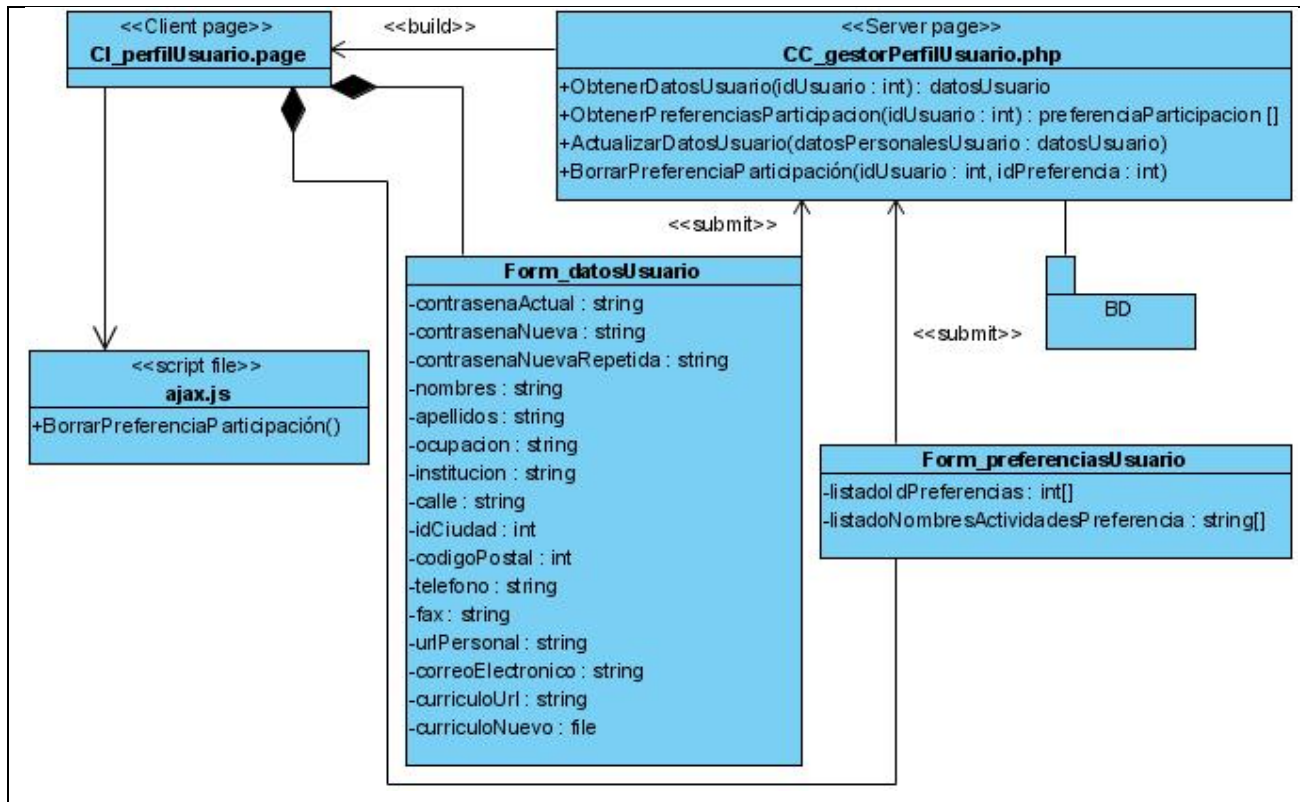


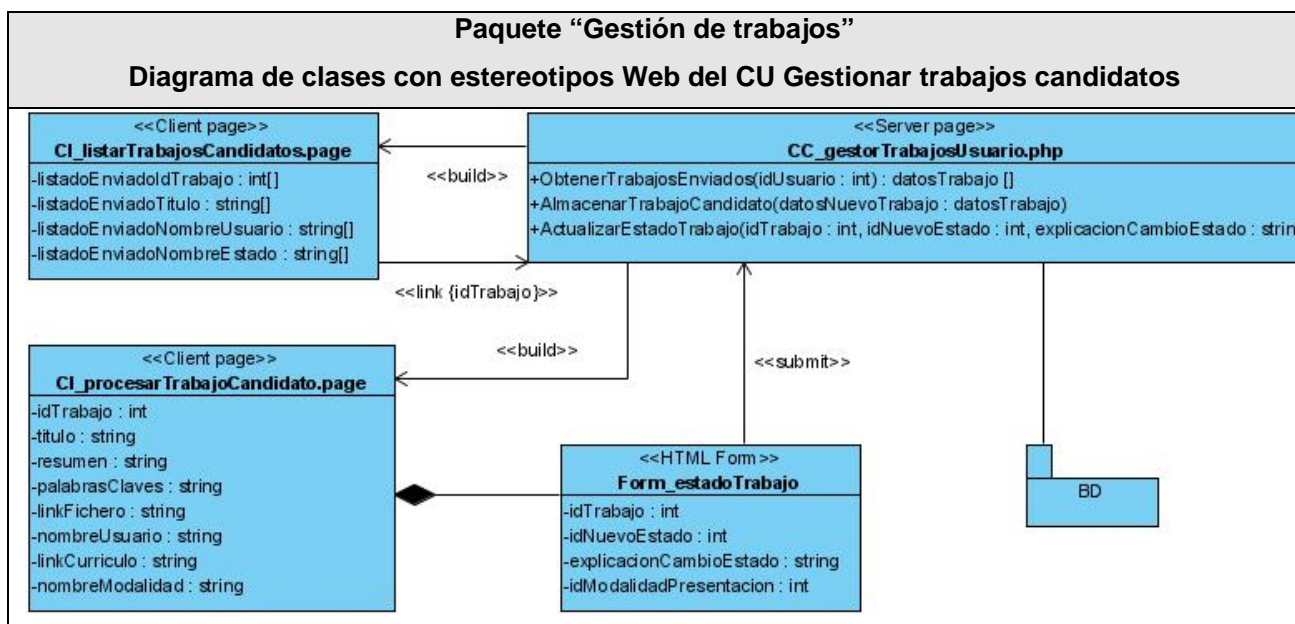
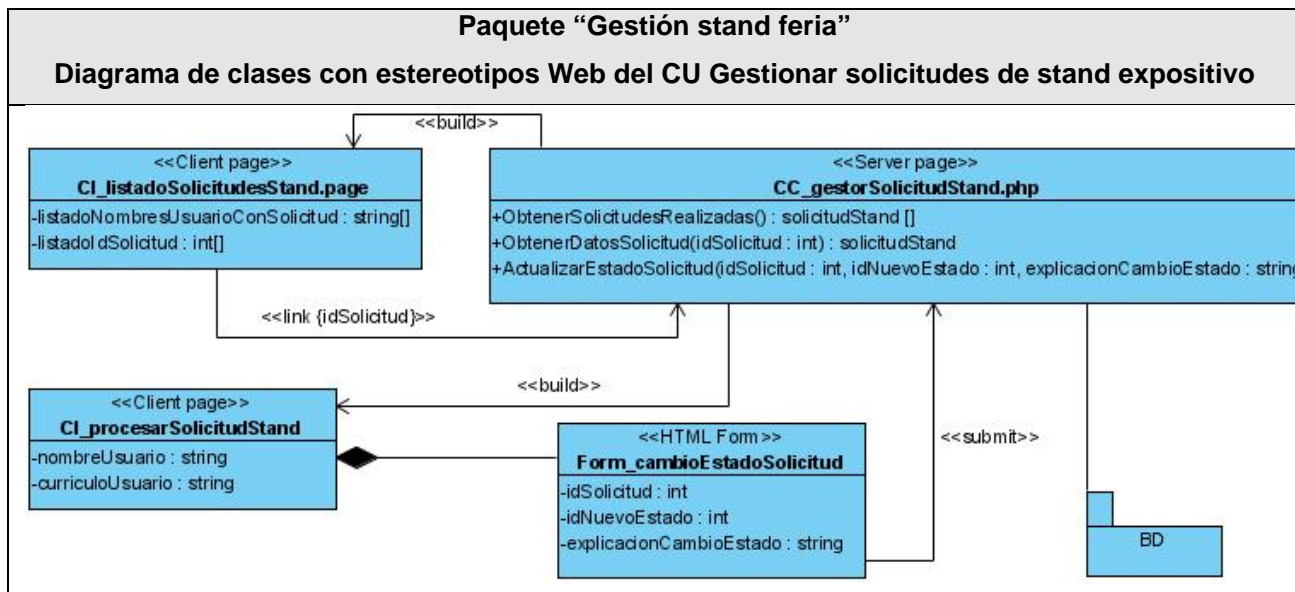


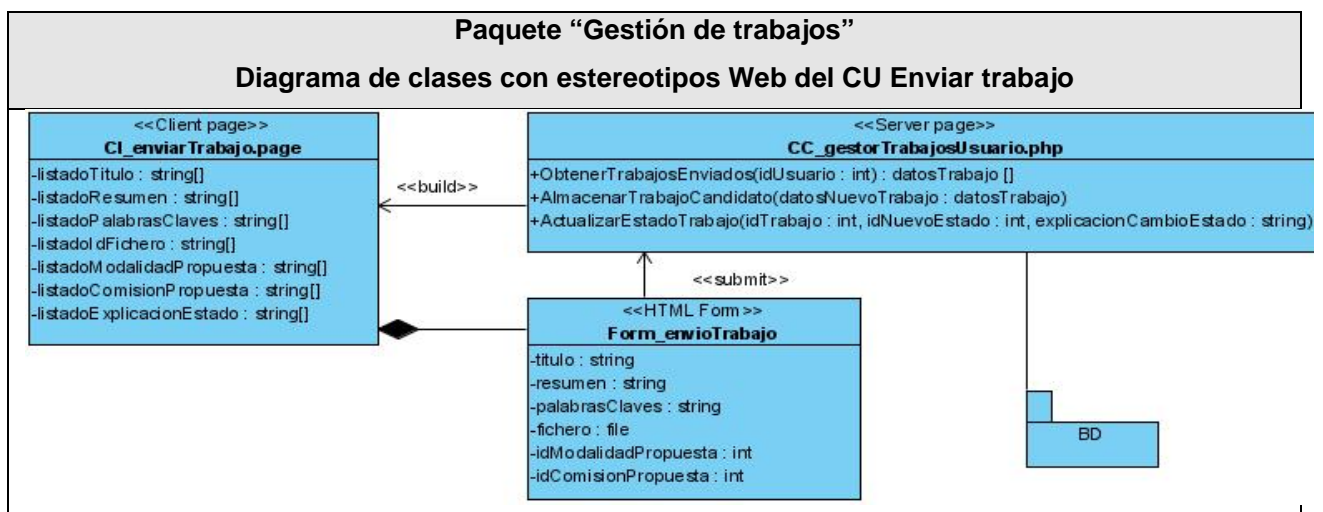
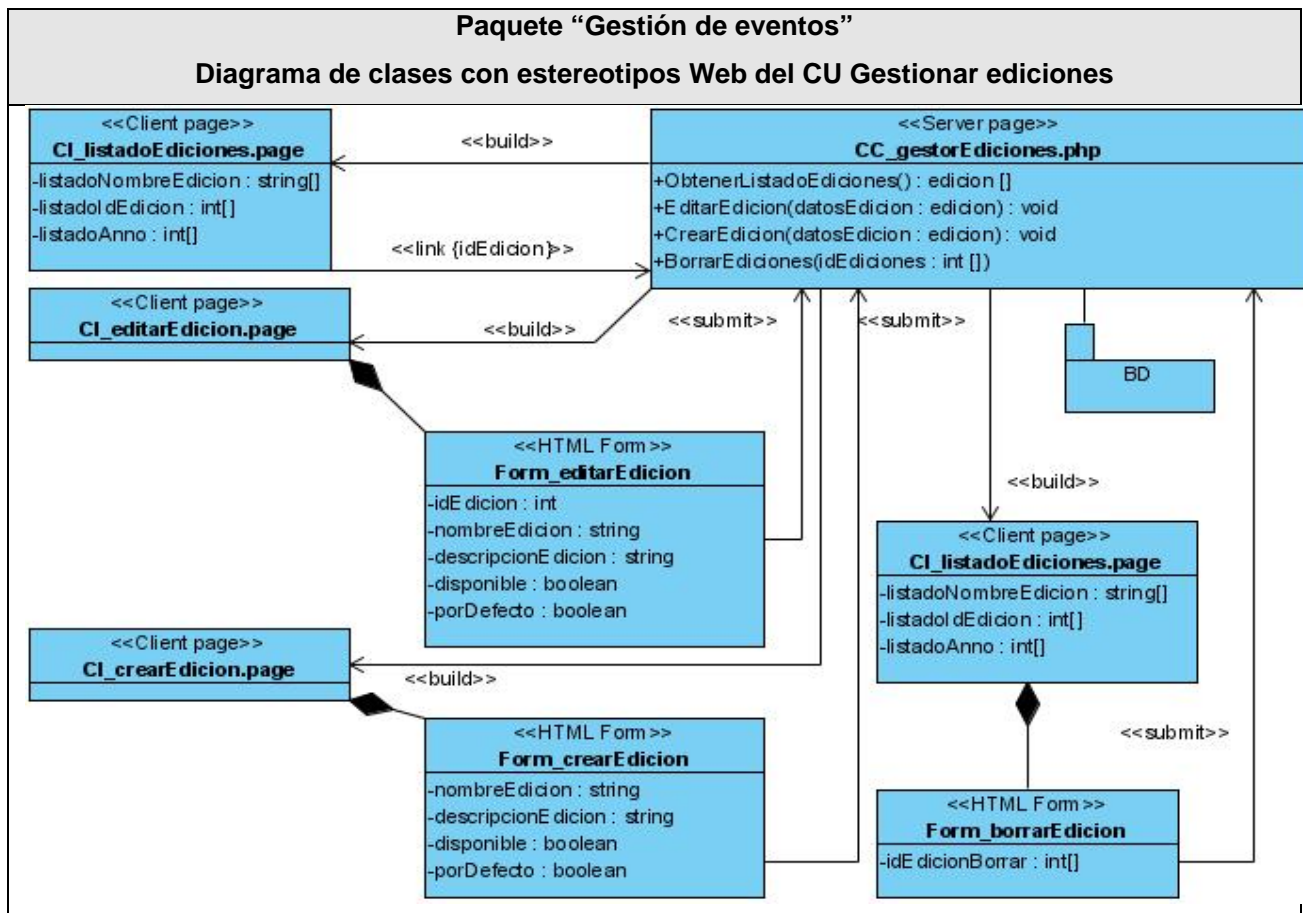
Anexo 4. Diagramas de clases de diseño











Anexo 5. Diagrama entidad relación

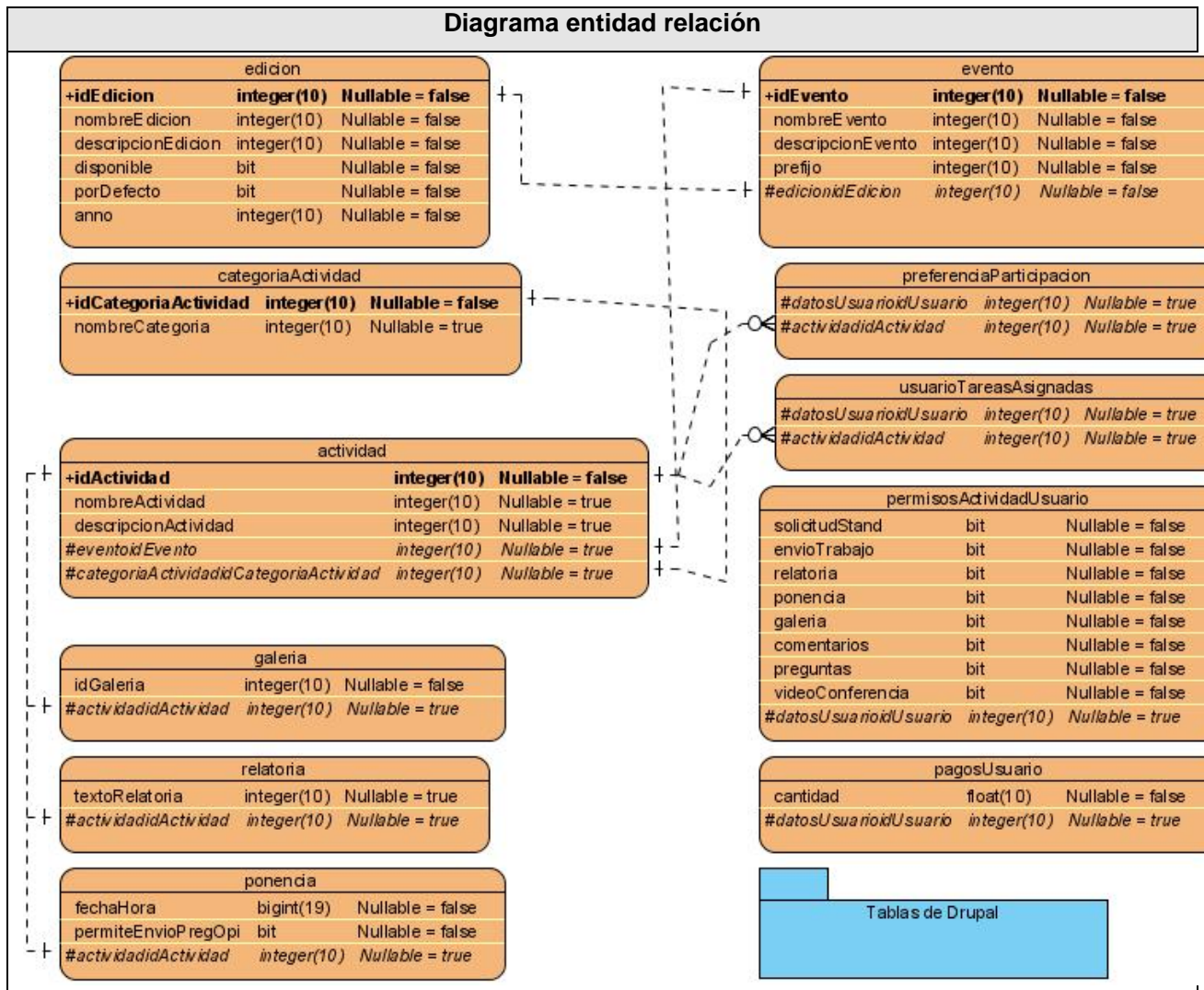
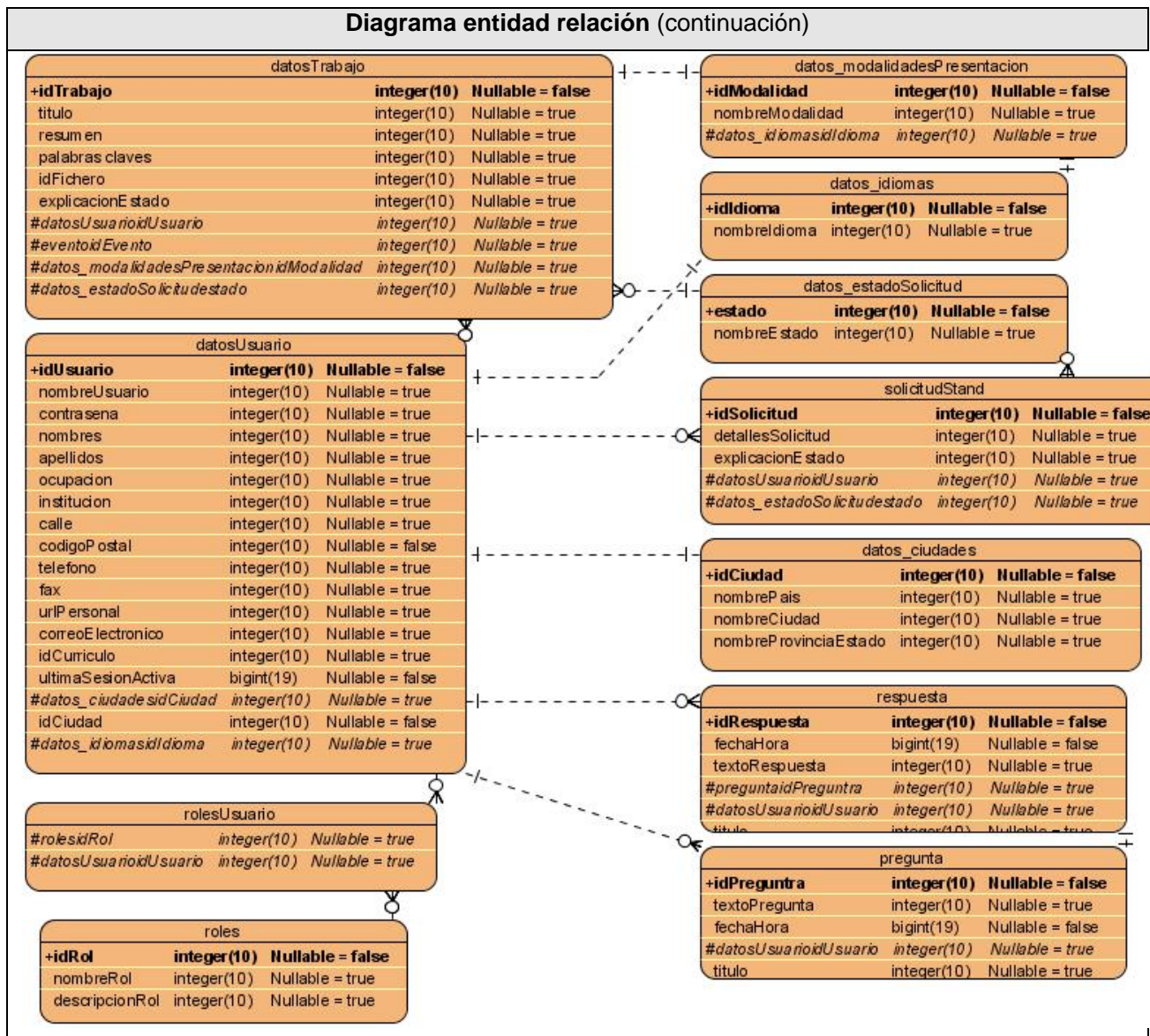
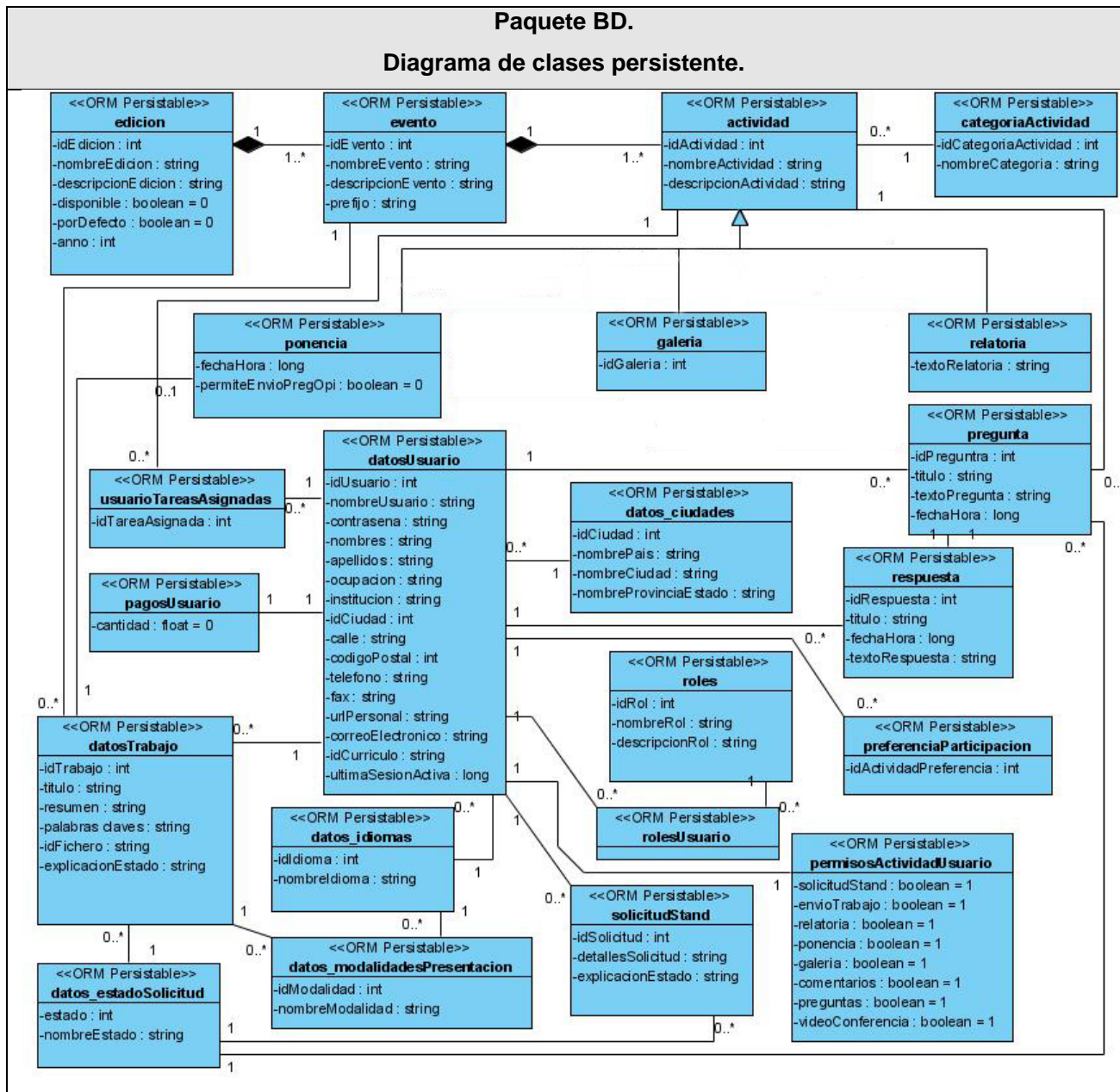


Diagrama entidad relación (continuación)



Anexo 6. Paquete BD. Diagrama de clases persistente.



Glosario de términos

- **AJAX.** Asynchronous JavaScript And XML. Javascript Asíncrono y XML. Es una técnica de desarrollo Web para crear aplicaciones interactivas, en las que se puede enviar y recibir información de los servidores sin necesidad de recargar las páginas completamente.
- **Apache.** En informática es el nombre de un software servidor de aplicaciones de código abierto.
- **Aplicación compleja.** Expresión que se utiliza para referirse al software que requiere de mucho tiempo para su elaboración.
- **Aplicaciones de escritorio.** Se le denomina a los software que se ejecutan en la maquina de los clientes, donde no hay interacción cliente-servidor.
- **Aplicaciones distribuidas.** Aplicaciones Web compuesta por distintos componentes que se ejecutan en entornos de ejecución separados.
- **Aplicaciones Web.** Software que se utiliza en una red accediendo a servidores de aplicaciones que brindan determinados servicios.
- **Applet.** Programa pequeño diseñado para ser ejecutado desde otras aplicaciones, comúnmente incrustados a los navegadores Web.
- **Arquitectura de software.** Disciplina dedicada a definir los parámetros estructurales del software.
- **Balance de carga.** Es un concepto usado en informática que se refiere a la técnica usada para compartir el trabajo a realizar entre varios recursos, como procesos, ordenadores, discos u otros.
- **Captcha.** Acrónimo de Completely Automated Public Turing test to tell Computers and Humans Apart, (Prueba de Turing pública y automática para diferenciar a máquinas y humanos). Es una técnica utilizada para evitar que programas mal intencionados puedan utilizar servicios informáticos y provocar problemas en estos, ya que consiste en reconocer un conjunto de caracteres distorsionados.
- **CASE.** Computer Aided Software Engineering. Ingeniería de Software Asistida por Ordenador. Categoría a la que pertenecen todas las aplicaciones destinadas a facilitar el trabajo de modelado a los Ingenieros de software.
- **Caso de uso.** En ingeniería del software, un caso de uso es una técnica para la captura de requisitos.
- **CGI.** Common Gateway Interface. Es una importante tecnología de la World Wide Web que permite a un cliente (a través de su navegador Web) solicitar datos de un programa ejecutado en un servidor Web.

- Clase. En POO se utiliza para definir las propiedades y funcionalidades de un elemento determinado.
- CMS. Content Managed System. Sistemas de Gestión de Contenido. Sistemas que facilitan la creación y administración de contenidos en la Web, que permiten manejar de manera independiente el contenido y el diseño.
- Componente. En informática son aplicaciones que no funcionan por si solas sino que interactúa con otra aplicación para aportarle una función o utilidad específica.
- Comunidades de desarrollo. Término por el que se denomina a agrupaciones de personas que se dedican a realizar un trabajo en común, que habitualmente cuentan en Internet con sitios Web donde propician la participación y aclaran dudas sobre el trabajo que realizan.
- Confirmación de correo. Técnica utilizada para verificar que los usuarios introducen direcciones de correos electrónicos existentes y de su propiedad.
- Controladores de versiones. Herramienta colaborativa que permite a los equipos de desarrollo gestionar los cambios en los producto de software.
- CSS. Cascade Style Sheets. Hojas de Estilo en Cascada. Es un lenguaje formal usado para definir la presentación de un documento estructurado con HTML.
- Curva de aprendizaje. Es un diagrama que se utiliza para describir el grado de éxito obtenido durante el aprendizaje de determinado tema, en el transcurso del tiempo.
- DDL. Data Definition Language. Lenguaje de Definición de Datos. Es un lenguaje proporcionado por los SGBD que permite a los usuarios definir las estructuras que almacenarán los datos, así como de los procedimientos para consultarlos.
- Debian GNU/Linux. Una de las distribuciones Linux sin ánimo de lucro más popular del mundo.
- Disparadores. En informática se utiliza para referirse a un evento que se ejecuta cuando se cumple una condición establecida al realizar una operación de inserción, actualización o borrado en campos de una base de datos.
- DML. Data Manipulation Language. Lenguaje de Manipulación de Datos. Es un lenguaje proporcionado por los SGBD que permite a los usuarios de los mismos llevar a cabo las tareas de consulta o manipulación de los datos.
- Encapsulamiento. Técnica de programación que consiste en reunir a todos los elementos que pueden considerarse pertenecientes a una misma entidad.
- Evento. Empleado en la POO para referirse a la reacción que puede provocar un cambio en un objeto.

- **Framework.** En informática es una estructura de soporte definida en la cual otro proyecto de software puede ser organizado y desarrollado.
- **Fundación Mozilla.** Es una organización sin ánimo de lucro dedicada a la creación de software libre. Dedicada a mantener la elección y la innovación en Internet. Ejemplos de su trabajo lo constituyen Mozilla Firefox, Mozilla Thunderbird y Bugzilla.
- **GNU.** Acrónimo recursivo de “Gnu no es Linux” (del inglés “Gnu No Unix”).
- **GNU GPL.** GNU General Public License. Licencia Pública General GNU. Es una licencia creada por la Free Software Foundation orientada principalmente a proteger la libre distribución, modificación y uso del software.
- **Google Trend.** Es una herramienta desarrollada por la compañía Google que muestra los términos de búsqueda más populares de Internet.
- **Groupware.** Sistemas de trabajo colaborativo. Programas informáticos que integran el trabajo de varios usuarios, facilitando la labor de los equipos.
- **Herencia.** Vocablo usado en la informática para agrupar y organizar clases con propiedades y funcionalidades en común.
- **HTML.** HyperText Markup Language. Lenguaje de Marcado de Hipertexto. Lenguaje estándar utilizado en Internet para dar estructura a la información y presentarla en forma de hipertexto.
- **HTTP.** HyperText Transfer Protocol. Protocolo de transferencia de hipertexto. Protocolo utilizado en la Web.
- **IDE.** Integrated Development Environment. Entorno de Desarrollo Integrado. Son un tipo de editores de texto realizados especialmente para programar en los distintos lenguajes, que proveen de herramientas de compilación, completamiento de código, y otros valores agregados.
- **IIS.** Internet Information Server. Servidor de Información de Internet. Software servidor de aplicaciones desarrollado por Microsoft.
- **IBM.** International Business Machines. Empresa que fabrica y comercializa herramientas, programas y servicios relacionados con la informática.
- **Ingeniería de software.** Disciplina que designa el conjunto de técnicas destinadas a la producción de software, más allá de la sola actividad de programación de los mismos.
- **Instancia de una clase.** En POO se utiliza para referirse a un Objeto, que es una entidad que tiene un conjunto de atributos y funcionalidades.
- **Interfaz.** Término utilizado en la informática para referirse a un elemento que sirve de conexión a otros dos independientes.

- Interbase. SGBD desarrollado y comercializado por la compañía Borland Software Corporation.
- Internauta. Usuario de una red informática de comunicación internacional, cibernauta.
- Interprete PHP. Es un programa capaz de analizar código PHP y ejecutar las instrucciones que en este se dan a medida que sea necesario.
- ISAPI. Internet Server Application Programming Interface. Interfaz de Programación de Aplicaciones de Servidor de Internet. Es un conjunto de APIs que se utilizan con el fin de crear extensiones y filtros de servidor OLE para servidores Web como Servicios de Internet Information Server.
- JavaScript. Lenguaje de programación interpretado, utilizado en las páginas Web, con una sintaxis semejante a la del lenguaje Java y el lenguaje C.
- LDAP. Lightweight Directory Access Protocol. Protocolo a nivel de aplicación que permite el acceso a un servicio de directorio para buscar diversa información en un entorno de red, habitualmente usuario y contraseña para la autenticación en sistemas.
- Lenguaje de programación C. Es uno de los lenguajes de programación más populares para crear software de sistemas, ya que con él se pueden lograr funcionalidades de sistema a bajo nivel.
- Lenguaje de programación interpretado. En informática se utiliza para referirse a los lenguajes de programación diseñados para ser ejecutado por medio de un intérprete, también conocidos como lenguajes script.
- Lenguaje de programación Perl. Lenguaje de programación de scripts multiplataforma muy utilizado para realizar actividades de procesamiento de texto.
- Licencia BSD. Licencia Berkeley Software Distribution. Licencia que pertenece al grupo del software libre.
- Licencia privativa. Nombre por el que son conocidas las licencias de software que restringen a los usuarios en el uso del mismo, en cuanto a distribuirlo o modificarlo, entre otras restricciones.
- Listas de distribución de correo. Se conoce por esto al envío de información vía correo electrónico a personas interesadas en un tema, en algunos casos los usuarios inscritos en estas listas son los mismos que distribuyen dicha información.
- MediaWiki. Es un motor para desarrollar wikis bajo licencia GNU GPL, programado en PHP usando MySQL.
- Mensaje. Utilizado en la POO para referirse a la comunicación dirigida a un objeto, que le ordena que ejecute uno de sus métodos con ciertos parámetros.

- Moodle. CMS Open Source que ayuda a los educadores a crear comunidades de aprendizaje en línea.
- MySQL. Software de gestión de bases de datos relacionales, bajo licencia GNU GPL.
- Navegador Web. Programa o aplicación informática que se usa para navegar por las redes informáticas y acceder a documentos, imágenes y demás información, fundamentalmente usando el protocolo HTTP.
- Open Source. Código abierto. Término con el que se conoce al software distribuido y desarrollado libremente. Fue utilizado por algunos usuarios de la comunidad del software libre, como reemplazo al ambiguo nombre original en inglés del software libre, free software.
- Página Web dinámica. Nombre que se le da a las páginas Web que son generadas por los servidores de aplicaciones según condiciones o acciones específicas del usuario que este accediendo a la misma.
- Patrones GRASP. General Responsibility Assignment Software Patterns. Patrones de Software para la Asignación General de Responsabilidades. Son una serie de patrones de software basados en la experiencia de expertos, para desarrollar software más robusto, reutilizable y libre de errores de diseño.
- PHP-Nuke. CMS desarrollado con PHP y MySQL.
- PhpPgAdmin. Herramienta elaborada usando PHP para gestionar el SGBD PostgreSQL mediante el uso de páginas Web.
- Plug-in. En informática son aplicaciones que no funcionan por si solas sino que interactúa con otra aplicación para aportarle una función o utilidad específica.
- Polimorfismo. Es la propiedad que tienen objetos de un mismo tipo, de comportarse de formas distintas.
- POO. Programación Orientada a Objetos. Es un paradigma de la programación que proporciona un medio bien definido para visualizar el dominio de un problema e implementar en un lenguaje de programación la solución a ese problema, utilizando herencia, modularidad, polimorfismo, y encapsulamiento.
- PostNuke. CMS libre y gratuito derivado de PHP-Nuke.
- Procedimiento almacenado. Código SQL almacenado físicamente en una base de datos que se utiliza para trabajar con los datos sin necesidad de conocer la estructura interna de la base de datos.
- Release. Versión funcional de un sistema.

- Servidor de aplicaciones. En Informática se denomina un servidor en una red de computadoras que ejecuta ciertas aplicaciones, como lo son los servidores de Apache, Internet Information Server.
- Servidor. Computador destinado a compartir algún recurso.
- SGBD. Sistemas de Gestión de Bases de Datos.
- Sintaxis. Disciplina encargada de estudiar las formas en que se combinan las palabras, utilizada en informática para describir las formas correctas de usar los lenguajes de programación.
- Sistema Web. Los sistemas Web son conocidos también como aplicaciones Web, denominación que se le da al Software que se utiliza en una red accediendo a servidores de aplicaciones que brindan determinados servicios.
- Sistemas de respaldo. Son herramientas que permiten hacer copias de información sensible, con la finalidad de poder restaurar los datos en caso de pérdida de estos.
- Smarty. Es un motor de plantillas para PHP, utilizado para separar el código PHP del HTML.
- Soporte comercial. Ayuda personalizada sobre un sistema, brindada por sus creadores, generalmente pagada, base de las ganancias del software libre.
- SQL. Structured Query Language. Lenguaje de Consulta Estructurado. Es un lenguaje de acceso a bases de datos que permite realizar operaciones sobre las mismas.
- Sybase IQ. Es un motor de bases de datos desarrollado por la empresa Sybase.
- Tecnología. Conjunto de los conocimientos propios de una técnica.
- Texto plano. Utilizado para referirse al contenido de los ficheros, donde la información que se almacena no tiene formato.
- Tiempo de corrida. Lapso de tiempo en que un software se encuentra en ejecución.
- Ubuntu. Es una distribución Linux, basada en Debian GNU/Linux, que ofrece un sistema operativo predominantemente enfocado a computadoras de escritorio aunque también proporciona soporte para servidores.
- UML. Unified Model Language. Lenguaje de modelado unificado.
- URL. Uniform Resource Locator. Localizador uniforme de recurso. Es un estándar para referenciar recursos en la Web, donde cada elemento tiene una dirección única, según su localización.
- Versión. Número que indica el nivel de desarrollo de un programa.
- Vistas. En informática se emplea para referirse a una consulta SQL de una o varias tablas, también se les considera una tabla virtual.
- Visual Basic Script. Lenguaje de programación interpretado desarrollado por Microsoft, utilizado en las páginas Web.

- W3C. World Wide Web Consortium. Consorcio dirigido por Berners-Lee, en el cual desarrollan y mantienen los estándares que permiten a los ordenadores de la Web almacenar y comunicar efectivamente diferentes formas de información.
- Wiki. Es una aplicación Web colaborativa que permite a sus usuarios crear, modificar o borrar los contenidos de la misma.
- Wikipedia. Es una enciclopedia Web de uso libre, basada en tecnología Wiki.
- WordPress. CMS enfocado a la creación de blogs en Internet.
- XML. eXtensible Markup Language. Lenguaje de Marcas de Extensible. Es un metalenguaje extensible de etiquetas desarrollado por el W3C.

Bibliografía

- (MIC), M. D. I. Y. L. C. Historia de la Convención y Feria Internacional Informática, 2007. [2007].
 Disponible en: http://www.informaticahabana.com/?q=es/hist_es
- (MIC), M. D. L. I. Y. L. C. D. C. Convocan a próxima cita mundial Informática 2009, 2007. [2007].
 Disponible en: <http://www.informaticahabana.com>
- (UCI), U. D. L. C. I. Diseño metodológico de la investigación científica, 2006. [2007]. Disponible en:
http://tesis.uci.cu/viewpage.php?page_id=21
- (UCI), U. D. L. C. I. Diseño teórico de la investigación científica, 2006. [2007]. Disponible en:
http://tesis.uci.cu/viewpage.php?page_id=20
- AB, M. MySQL 5.0 Reference Manual, 2008. [Disponible en:
<http://dev.mysql.com/doc/refman/5.0/es/index.html>
- AMBLER, S. W. The Object Primer. 3rd Edition. Cambridge University Press, 2004. p. 0-521-54018-6
- BENC. Why Choose Drupal?, 2006. [Disponible en:
http://www.digitalsolutions.ph/couchkamotereviews/why_choose_drupal
- GLOWIAK, M. MySQL vs. PostgreSQL. Disponible en:
http://monstera.man.poznan.pl/wiki/index.php/MySQL_vs_postgres
- GNU.ORG. Diversas licencias y comentarios sobre ellas, 2006. [Disponible en:
<http://www.gnu.org/licenses/license-list.es.html>
- LIGHTTPD, E. lighttpd fly light, 2008. [Disponible en: <http://www.lighttpd.net/>
- MORALES, P. C. Desarrollo de Aplicaciones Distribuidas basadas en Tecnologías Web, 1999.
 [Disponible en: <http://trevinca.ei.uvigo.es/~pcuesta/publicaciones/TecWeb.pdf>
- PECOS, D. PostgreSQL vs. MySQL. Disponible en:
http://www.netpecos.org/docs/mysql_postgres/index.html
- WELICKI, L. Patrones y Antipatrones: una Introducción - Parte I. Disponible en:
http://www.microsoft.com/spanish/msdn/comunidad/mtj.net/voices/MTJ_2864/default.aspx
- (IEBS), I. D. E. B. S. *Eventos - Sesiones virtuales - Instituto de Empresa Business School*, 2005.
 [2008]. Disponible en: <http://www.ie.edu/eventos/virtual.asp>
- (MIC), M. D. I. Y. L. C. *Evento Virtual Informática 2007*, 2007. [2007]. Disponible en:
http://www.informaticahabana.com/evento_virtual/
- AGUILAR, V. and P. SUAUI. *MySQL vs. PostgreSQL*. Disponible en: <http://www.fedora-es.com/node/189>
- ALEXANDER, C. *The Timeless Way of Building*. Oxford University Press, USA (1979), 1979. 576 p.
- AMBLER, S. W. *The Elements of UML 2.0 Style*. Cambridge University Press, 2005. p. 0-521-61678-6
- BENC. *Why Choose Drupal?*, 2006. [Disponible en:
http://www.digitalsolutions.ph/couchkamotereviews/why_choose_drupal
- CIVE. *Congreso Internacional Virtual de Educación*. Disponible en:
<http://www.steiformacio.com/cive/cive08.asp>
- COLLECTIVE, O. S. *OpenSourceCMS*. Disponible en: <http://www.opensourcecms.com/>

- COMPANY, A. *AOL Web Site*. Disponible en: <http://corp.aol.com/>
- COMPANY, T. *TIOBE Programming Community Index for April 2008*. Disponible en: <http://www.tiobe.com/index.php/content/paperinfo/tpci/index.html>
- CORAI, R. *Drupal es el mejor CMS del 2007*. Disponible en: <http://www.revistacms.com/content/view/106/34/>
- CORPORATION, P. B. *CMS Matrix*. Disponible en: <http://www.cmsmatrix.org/>
- EUMED, G. D. I. *Encuentros Académicos Internacionales*. Disponible en: <http://www.eumed.net/eve/index.htm>
- FERNANDEZ, S. P. *Comparativa Servidores Web*. Disponible en: <http://observatorio.cnice.mec.es/modules.php?op=modload&name=News&file=article&sid=363>
- FUNDATION, M. *Mozilla Foundation Web Site*. Disponible en: <http://www.mozilla.org/>
- GAMMA, E.; R. HELM, et al. *Design patterns : elements of reusable object-oriented software*. 5ta. Addison Wesley, 1995. 416 p. 0201633612
- GROUP, P. G. D. *PostgreSQL: The world's most advanced open source database*. Disponible en: <http://www.postgresql.org/>
- HAREM. *Lighttpd Vs. Apache*, PhpMagazine. Disponible en: http://www.phpmagazine.net/2007/02/lighttpd_vs_apache.html
- HÉGARET, P. L. *Document Object Model (DOM)*, 2008. [Disponible en: <http://www.w3.org/DOM/faq.html#DHTML-DOM>
- IEEE-SA. *IEEE Std 1471-2000 IEEE Recommended Practice for Architectural Description of Software-Intensive Systems*, 2004. [Disponible en: http://standards.ieee.org/reading/ieee/std_public/description/se/1471-2000_desc.html
- IFEMA. *SIMO Feria Internacional de Informática, Multimedia y Comunicaciones*. Disponible en: <http://www.ifema.es/ferias/simo/default.html>
- JACOBSON, I.; G. BOOCH, et al. *El proceso unificado de desarrollo de software*. Addison-Wesley, 2000. 458 p. *The Addison-Wesley object technology series*. 84-7829-036-2
- LTD, P. P. *Drupal Wins Overall 2007 Open Source CMS Award*, Packt Publishing Ltd. Disponible en: <http://www.packtpub.com/article/drupal-wins-overall-2007-open-source-cms-award>
- MERCHANDISING, P. *Eventos Virtuales - La Empresa*, 2008]. Disponible en: <http://www.eventosvirtuales.com.ar/empresa.php>
- NASA. *NASA Web Site*. Disponible en: <http://appel.nasa.gov/>
- O'REILLY, T. *Programming Language Trends*. Disponible en: <http://radar.oreilly.com/archives/2006/08/programming-language-trends.html>

RESEARCH, Y. *Yahoo Research Web Site*. Disponible en: <http://research.yahoo.com/>

REYNOSO, C. B. *Introducción a la Arquitectura de Software*, 2006. [Disponible en: http://www.microsoft.com/spanish/msdn/arquitectura/roadmap_arq/intro.mspix

SAIEDIAN, H. *Una evaluación del modelo entidad relación extendido*, Departamento de Ciencias de la Computación, Universidad de Nebraska de Omaha. NE 68182-0500, USA. Disponible en: <http://www.dcc.uchile.cl/~cguierr/cursos/BD/extendido.pdf>

SANCHEZ, M. A. M. *Metodologías de desarrollo de software*. Disponible en: http://www.informatizate.net/articulos/metodologias_de_desarrollo_de_software_07062004.html

UBUNTU. *Proyecto Ubuntu*. Disponible en: <http://www.ubuntu-es.org/>

W3SCHOOLS, A. *Browser Statistics*. Disponible en: http://www.w3schools.com/browsers/browsers_stats.asp

WEITZMAN, L.; A. LEWIS-BOWEN, *et al.* *Using open source software to design, develop, and deploy a collaborative Web site*. Disponible en: <http://www-128.ibm.com/developerworks/ibm/library/i-ospace1/index.html?ca=drs->