

Universidad de las Ciencias Informáticas

Facultad 10



Título: Módulo para la Gestión Estadística de Drupal.

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

Autor(es): Yeisel José Rosales Ávila
Deybis García Rodríguez

Tutor(es): Ing .Yanier Castellanos González

Ciudad de la Habana, junio del 2007

Agradecimientos:

*Agradecemos a todos los que de alguna forma han contribuido a nuestra formación,
especialmente agradecemos:*

A nuestros padres por haber dado la fuerza y voluntad para graduarnos como ingenieros.

A nuestro tutor Yanier por su ayuda durante toda la tesis.

A nuestros amigos.

Nuestros compañeros de aula por convivir los cinco años de la carrera en la Universidad.

*A los muchachos del proyecto por su colaboración, muestras de confianza y ayuda
desinteresada.*

*A los señores oponentes por sus magníficas oponencias durante la predefensa, que fueron de
gran ayuda*

Nuestros familiares que son nuestra razón de ser.

A la Revolución que sin ella nada de esto hubiese sido posible.

Yeisel

Mis padres por tenerme presente en todo momento y ser parte de ellos.

A mi hermanos para que le sirva de ejemplo en etapas futuras...

A mi novia por estar siempre en mi corazón.

A mi familia por todo el tiempo que les he robado, darme su apoyo y confiar en mí.

*Ningún resultado que obtenga podrá devolverles a ellos todo el tiempo que les he robado.
A mi amigos especialmente a blanco, deybis, pocho, wendysh, yuri, el papa, yaima, yohan.*

*A mis compañeros de apartamento durante estos cinco años de estudio en la Universidad.
Con palabras no se puede expresar el significado de familia, amigos y seres queridos.*

Deybis

A mis padres, por el amor, apoyo y ejemplo que me han dado siempre, y por la confianza que depositaron en mí.

A mis hermanos Dannis y Lummis que siempre se mantuvieron aconsejándome y apoyándome en todo momento.

A Gallega que a pesar de no ser parte de mi familia siempre a estado al tanto de mis resultados.

A mis amigos de la Universidad y en especial a Wendysh y Yeisel.

A toda mi familia en general.

Introducción	1
Capítulo 1: Fundamentación Teórica.....	6
1.1 Los sistemas de gestión de contenidos. Drupal.....	6
1.1.1 APIs de Drupal.....	10
1.2 Servidor de aplicaciones. Apache.....	11
1.3 Metodología: Rational Unified Process (RUP).....	12
1.3.1 Principales características:	12
1.3.2 Lenguaje de Modelaje Unificado (UML).	14
1.3.3 Herramientas CASE – UML	16
1.3.4 Características del Visual Paradigm:	17
1.4 PHP	18
1.5 HTML	20
1.6 MYSQL.....	21
Capítulo 2. Características del sistema.	23
2.1 Problema.	23
2.2 Objeto de Automatización	24
2.3 Información que se maneja	24
2.4 Propuesta del sistema	24
2.5 Modelo del Dominio	25
2.6 Representación del modelo del dominio.....	25
2.7 Requisitos.....	25
2.8 Casos de Uso del sistema.....	27
2.8.1 Determinación y justificación de los actores del sistema.	27
2.8.2 Descripción y expansión de los casos de uso.....	28
2.9 Diagramas de caso de uso del sistema.	29
2.10 Casos de Uso expandidos.....	29
Capítulo 3. Análisis y diseño del sistema.	41
3.1 Análisis del sistema.....	41
3.1.1 Modelo conceptual de las clases de análisis.	41
3.2 Análisis del diseño.	41
3.2.1 Diagramas de interacción.....	41
3.2.2 Descripción de las clases de análisis.....	42
3.3 Diagrama de clases.....	51
3.4 Modelo de datos	51
3.5 Diagrama Entidad Relación de la BD.....	52
3.5.1 Descripción de las tablas.....	53
Capítulo 4: Implementación y Prueba.....	56
4.1 Diagrama de componentes.....	56
4.2 Prueba.....	56
4.2.1 Normas de prueba	57
Conclusiones generales.....	60

Recomendaciones.....	61
Bibliografía	62
Anexos	64
Anexo 1: Modelo del dominio.	64
Anexo 2: Modelo de Casos de uso del sistema.	65
Anexo 3: Modelo de análisis.....	66
Anexo 4: Diagramas de colaboración.	70
Anexo 5: Diagrama de clases.....	75
Anexo 6: Diagrama Entidad-Relación.	76
Anexo 7: Diagrama de despliegue.	77
Anexo 8: Diagrama de Componentes.	78
Glosario de términos	79

Introducción

Normalmente las estadísticas son el resultado de un análisis de datos recuperados de manera histórica las mismas se obtienen a través de líneas de código que se colocan en cada documento del sitio aunque también pueden colocarse en un servidor así podrá saber que documentos son visitados por un mismo usuario, en que punto sale del sitio y que tiempo permanece en el mismo.

Las estadísticas de una web se ofrecen con las siguientes ventajas:

- Reporte detallado de las visitas a su sitio en tiempo real.
- Estadísticas comprensibles, fáciles de entender.
- No necesita descargar programas ni instalar nada en su computadora.
- Solo tiene que registrarse, pegar un par de líneas de código en su página y de inmediato comenzará a recibir estadísticas.
- No se necesitan datos personales, solo su nombre de usuario, contraseña y la dirección del sitio para seguir las estadísticas.
- Si tiene dudas de como usar este servicio, se ofrece ayuda personalizada.
- Las estadísticas de su sitio pueden ser públicas o solo privadas para usted.

La mayoría de las herramientas de análisis de los sitios web proporcionan respuestas a las siguientes preguntas:

- ¿Cuáles son los caminos más comunes a las páginas más importantes del sitio web?
- ¿Qué palabras claves generan la mayor parte del tráfico del sitio web a partir de los buscadores?
- ¿Cuánto tiempo pasan los visitantes del país en el sitio web?
- ¿Cuántas páginas ven normalmente los visitantes del sitio web?
- ¿Cuántos visitantes proceden de cada región importante del sitio web?
- ¿Qué porcentaje del total suponen los visitantes?
- ¿Cuántos nuevos usuarios visitan el sitio cada mes?

¿Por que son importantes las estadísticas en la actualidad?

Las estadísticas de un sitio web son fundamentales para lograr el éxito, usted debe saber de donde vienen sus visitas, cuantas visitas tienes en el día, de que país proceden, que navegador utiliza, sistema operativo, saber la posición del usuario dentro de la web al instante, las páginas que visita cada usuario, el tiempo que pasa en cada página y que página de su sitio web es la más visitada (1).

Hay que darle importancia necesaria a las estadísticas del sitio, de no hacerlo se estará perdiendo una de las mejores armas para lograr que el sitio web sea exitoso.

Drupal

Es un sistema de gestión de contenidos (Content Management System o CMS) que se utiliza principalmente para facilitar la gestión de la web, ya sea en Internet o en una Intranet. Está compuesto fundamentalmente por módulos que permiten su buen funcionamiento.

¿Que es un módulo?

Un módulo es un componente autocontrolado de un sistema, el cual posee una interfaz bien definida hacia otros componentes; algo es modular si es construido de manera tal que se facilite su ensamblaje, acomodamiento flexible y reparación de sus componentes.

Un programa puede constar de diferentes módulos y cada cual actúa independientemente del otro.

El módulo estadístico en particular permite recoger y visualizar (de forma gráfica y tabular) información estadística acerca del uso que hacen los usuarios de las publicaciones. Entre otra información:

- Visitas.
- Visualizaciones de páginas.
- Ampliación de páginas.
- Impresión de páginas.
- Click sobre ítems.
- Acceso a versión reducida de páginas.
- Seguimiento de la secuencia de acciones del usuario.

La implementación de módulos estadísticos varía en dependencia de la fuente de datos y de la naturaleza del sistema al que representan. En un ambiente web los módulos estadísticos están dirigidos mayormente al conteo de hits y visitas de los internautas con el objetivo de identificar tendencias de los usuarios así como popularidad del contenido que se publica. En este caso se pretende proveer de esta funcionalidad al CMS Drupal el cual está siendo utilizado en el proyecto Portales PHP de la facultad 10.

Existen varios módulos para el control estadístico de Drupal, pero estos no cuentan con las necesidades específicas del proyecto para resolver sobre todo los aspectos globales de navegación y datos del sistema así como la visualización gráfica de los mismos. Actualmente los recursos que generan el control estadístico de Drupal en cuestión de navegación al sitio en el proyecto no cumplen con las necesidades específicas del mismo. No se cuenta con un módulo estadístico que cumpla todas las prestaciones que se necesitan para su desarrollo. El proyecto ha venido desarrollando una serie de portales web y no hay un control de las estadísticas que cumpla con sus requerimientos, ejemplo los usuarios que visitan el sitio, estadísticas tales como el usuario que más ha visitado, mostrar gráficamente el % de visitas, navegador que se utiliza y sistema operativo utilizado por el mismo.

Una vez mostrada la situación a que se enfrenta y luego de hacer un breve análisis del estado del arte se está en condiciones de plantear la **problemática**.

- No disponer de un módulo basado en novedosos elementos para controlar, generalizar y explotar todos los movimientos y operaciones de un usuario dentro de la web.
- ¿Cómo se puede implementar un módulo estadístico que cumpla con las necesidades específicas del proyecto?

Como **novedad práctica** se tendrá que por primera vez en el proyecto se va a implementar un sistema de software en el cual se gestionen y administren los procesos estadísticos de Drupal que cumpla con las necesidades específicas del mismo. El **objeto de estudio** se basará en la extracción estadística de los sitios desarrollados sobre Drupal en sentido general, y como **objetivo** o meta a alcanzar se pretende desarrollar un módulo estadístico de Drupal en el proyecto productivo, para la gestión estadística de los sitios web que sean hijos de la utilización de este CMS y que ayude a mantener el control de cada usuario dentro del sitio en cuestión. A partir del objetivo general se plantean los siguientes objetivos específicos:

- Estudio del estado del arte del desarrollo de módulos estadísticos para la gestión estadística de Drupal.
- Implementar el modelo de análisis y diseño del sistema de gestión estadística para Drupal.
- Análisis de los resultados de la aplicación del sistema desarrollado.

Como todo proyecto en la actualidad se tiene un campo de acción “Gestión estadística de los sitios en Drupal” sobre el cual girará la problemática llegando así a la **hipótesis** del problema que está respaldada por la afirmación que si se implementa el sistema para la gestión estadística de Drupal en el proyecto, entonces se ayuda a resolver el control estadístico de una web que cumpla con las necesidades específicas del mismo y así lograr describir la actividad de cada usuario que visite el sitio. Para darle solución al problema se llevaron a cabo las tareas de investigación dentro de las cuales se pueden encontrar:

- Análisis de la herramienta para generar las estadísticas.
- Análisis de las herramientas según la arquitectura establecida.
- Análisis del CMS Drupal.
- Análisis de la APIs de Drupal.
- Análisis de la versión de Drupal a utilizar para el desarrollo.
- Pruebas :
 - Estrategias de pruebas.
- Ejecución y resultados de las pruebas. Análisis de resultados.
- Tratamiento de errores.

El presente trabajo consta de cuatro capítulos:

En el capítulo 1 se realiza un estudio del estado del arte acerca de las principales técnicas de programación existente a nivel mundial y en la universidad, además de un análisis sobre las tendencias, tecnologías, conceptos y metodologías utilizadas en la actualidad a nivel internacional, en el desarrollo de aplicaciones para el control estadístico.

El capítulo 2 se hace una descripción de la solución propuesta, se definen y describen cada uno de los conceptos del modelo del dominio y algoritmos utilizados para la solución que se propone. Así como una valoración crítica del sistema propuesto.

En el capítulo 3 se hace una valoración del análisis y diseño propuesto, se describen cada una de las clases que participan para la implementación del sistema.

En el capítulo 4 se hace referencia a la validación de la solución propuesta, con la evaluación y los resultados obtenidos.

Capítulo 1: Fundamentación Teórica

En el presente capítulo se realiza un estudio del estado del arte acerca de las técnicas de programación existente a nivel mundial, además de un análisis de las tendencias, tecnologías y metodologías utilizadas en la actualidad a nivel internacional, en el desarrollo de aplicaciones para el control estadístico de Drupal y en general en aplicaciones web, así como las plataformas de desarrollo que las soportan, fundamentándose en las seleccionadas para la solución que se propone.

1.1 Los sistemas de gestión de contenidos. Drupal.

Los sistemas de gestión de contenidos (Content Management System o CMS) son softwares que se utilizan principalmente para facilitar la gestión de la web, ya sea en Internet o en una intranet. Hay que tener en cuenta, sin embargo, que la aplicación de los CMS no se limita solo a la web.

Consisten en una interfaz que controla una o varias bases de datos donde se aloja el contenido del sitio. El sistema permite manejar de manera independiente el contenido por una parte y el diseño por otra. Así, es posible manejar el contenido y darle en cualquier momento un diseño distinto al sitio sin tener que darle formato al contenido de nuevo, además de permitir la fácil y controlada publicación en el sitio a varios editores.

El término Content Management System fue originalmente usado para la publicación de sitios web. Los primeros sistemas de administración de contenidos fueron desarrollados internamente por organizaciones que publicaban mucho en Internet, como revistas en línea, periódicos y publicaciones corporativas.

Estos son algunos de los puntos más importantes que hacen útil y necesaria la utilización de un CMS:

Inclusión de nuevas funcionalidades en la web. Esta operación puede implicar la revisión de multitud de páginas y la generación del código que aporta las funcionalidades. Con un CMS eso puede ser tan simple como incluir un módulo realizado por terceros, sin que eso suponga muchos cambios en la web. El sistema puede crecer y adaptarse a las necesidades futuras.

Mantenimiento de gran cantidad de páginas. En una web con muchas páginas hace falta un sistema para distribuir los trabajos de creación, edición y mantenimiento con permisos de acceso a las diferentes áreas. También se tienen que gestionar los metadatos de cada documento, las versiones, la publicación y caducidad de páginas y los enlaces rotos, entre otros aspectos.

Reutilización de objetos o componentes. Un CMS permite la recuperación y reutilización de páginas, documentos, y en general de cualquier objeto publicado o almacenado.

Páginas interactivas. Las páginas estáticas llegan al usuario exactamente como están almacenadas en el servidor web. En cambio, las páginas dinámicas no existen en el servidor tal como se reciben en los navegadores, sino que se generan según las peticiones de los usuarios. De esta manera cuando por ejemplo se utiliza un buscador, el sistema genera una página con los resultados que no existían antes de la petición. Para conseguir esta interacción, los CMS conectan con una base de datos que hace de repositorio central de todos los datos de la web.

Cambios del aspecto de la web. Si no hay una buena separación entre contenido y presentación, un cambio de diseño puede comportar la revisión de muchas páginas para su adaptación. Los CMS facilitan los cambios con la utilización, por ejemplo, del estándar CSS (Cascading Style Sheets u hojas de estilo en cascada) con lo que se consigue la independencia de presentación y contenido.

Consistencia de la web. La consistencia en un web no quiere decir que todas las páginas sean iguales, sino que hay un orden (visual) en vez de caos. Un usuario nota enseguida cuándo una página no es igual que el resto de las de la misma web por su aspecto, la disposición de los objetos o por los cambios en la forma de navegar. Estas diferencias provocan sensación de desorden y dan a entender que la web no la han diseñado profesionales. Los CMS pueden aplicar un mismo estilo en todas las páginas con el mencionado CSS, y aplicar una misma estructura mediante patrones de páginas.

El CMS Drupal ha llegado a ser mucho más que sólo un portal de noticias gracias a su arquitectura flexible. Se compone de una infraestructura base y un conjunto de módulos que ofrecen un amplio conjunto de funciones. Es posible añadir módulos de terceros para modificar el comportamiento de Drupal

u ofrecer nuevas funciones. Es un robusto sistema de ayuda online y páginas de ayuda para los módulos en núcleo tanto para usuarios como administradores, además de su código fuente totalmente abierto bajo los términos de la licencia GNU\GPL, el mismo puede ser adaptado según las necesidades de cada cual, es decir tanto el contenido como la presentación pueden ser individualizados de acuerdo a las preferencias definidas por el usuario los cuales se pueden registrar e iniciar sesión. Puede aplicarse en la construcción de una intranet. Drupal se puede integrar con un servidor LDAP, lo que posibilita la autenticación por el dominio en cualquier institución.

Los administradores de Drupal no tienen que establecer permisos para cada usuario. En lugar de eso, pueden asignar permisos a un 'rol' y agrupar los usuarios por roles.

El sistema de control de versiones de Drupal permite seguir y auditar totalmente las sucesivas actualizaciones del contenido: qué se ha cambiado, la hora y la fecha, quién lo ha cambiado, y más. También permite mantener comentarios sobre los sucesivos cambios o deshacer los cambios recuperando una versión anterior. El contenido creado en Drupal es, funcionalmente, un objeto (nodo). Esto permite un tratamiento uniforme de la información, como una misma cola de moderación para envíos de diferentes tipos, promocionar cualquiera de estos objetos a la página principal o permitir comentarios o no sobre cada objeto.

Plantillas (templates): El sistema de temas de Drupal separa el contenido de la presentación permitiendo controlar o cambiar fácilmente el aspecto del sitio web. Se pueden crear plantillas con HTML y/o con PHP.

Drupal incluye un potente agregador de noticias para leer y publicar enlaces a noticias de otros sitios web. Incorpora un sistema de caché en la base de datos, con temporización configurable.

Hay independencia de la base de datos. Aunque la mayor parte de las instalaciones de Drupal utilizan MySQL, existen otras opciones. Drupal incorpora una 'capa de abstracción de base de datos' que actualmente está implementada y mantenida para MySQL y PostgreSQL, aunque permite incorporar fácilmente soporte para otras bases de datos. Se utilizó MySQL ya que es una base de datos muy rápida en la lectura cuando utiliza el motor no transaccional MYISAM y muy manejable. En aplicaciones web hay baja concurrencia en la modificación de datos y en cambio el entorno es intensivo en lectura de datos, lo que hace a MySQL ideal para este tipo de aplicaciones.

Ha sido diseñado desde el principio para ser multiplataforma. Puede funcionar con Apache o Microsoft IIS como servidor web y en sistemas como Linux, BSD, Solaris, Windows y Mac OS X. Por otro lado, al estar

implementado en PHP, es totalmente portable. Toda la actividad y los sucesos del sistema son capturados en un 'registro de eventos', que puede ser visualizado por el administrador.

En cuanto al rendimiento Drupal incorpora un mecanismo de control de congestión que permite habilitar y deshabilitar determinados módulos o bloques dependiendo de la carga del servidor. Este mecanismo es totalmente configurable y ajustable. Tiene un mecanismo de caché elimina consultas a la base de datos incrementando el rendimiento y reduciendo la carga del servidor.

La administración y configuración del sistema se puede realizar enteramente con un navegador y no precisa de ningún software adicional.

En cuanto al análisis y seguimiento de las estadísticas Drupal puede mostrar en las páginas web de administración informes sobre referrals (enlaces entrantes), popularidad del contenido, o de como los usuarios navegan por el sitio.

Drupal es utilizado por desplegar sistemas como:

- Sitios web.
- Intranets.
- E-administración.
- Gestión de proyecto.
- Portales de comunidades.

Drupal suele ser criticado por todos aquellos que creen que la programación orientada a objetos (POO) es la mejor forma de diseñar la arquitectura del software y el hecho de no ver la palabra class (clase) en el código de Drupal hace parecer al mismo inferior ante otras soluciones.

El código de Drupal está separado en módulos, cada uno de los cuales define sus propias funciones, que además manejan la inclusión de archivos. La idea es que se cargue la menor cantidad de código por petición para que resulte lo más eficiente posible. En Drupal las funciones son por lo tanto definidas dentro de otras funciones en tiempo de ejecución. Pero PHP no permite este tipo de anidamiento con la

declaración de clases, eso significa que la inclusión de ficheros que definan clases debe estar en el nivel más alto, y no dentro de ninguna función.

Además Drupal usa en su diseño muchos otros paradigmas orientados a objetos, entre los esenciales: objetos, abstracción, encapsulamiento, polimorfismo, herencia, diseño de patrones.

1.1.1 APIs de Drupal.

A la hora de hacer cualquier módulo para darle funcionalidad a un sitio son muy utilizadas las APIs de Drupal, las cuales posibilitan desde la creación de un formulario hasta la visualización de imágenes. Hoy en día estas funcionalidades que trabajan en forma de hooks(ganchos) son muy utilizadas.

En definitiva, una nueva característica que proporcionará tremendas gratificaciones a aquellos desarrolladores de sitios montados sobre Drupal que deseen personalizar los formularios de la manera más sencilla posible, para dejar una interfaz limpia con sólo lo necesario en cada caso.

Los *hooks* son funciones de nombre *hook_xxx*, como *hook_menu*, *hook_perm* y *hook_help*. Los módulos las sobrescriben con funciones equivalentes de nombre *modulo_xxx*, como *indexer_menu*, *indexer_perm* e *indexer_help*. La función *hook_menu* es llamada siempre por Drupal antes de crear las páginas, antes de mandarlas al navegador, y en ella pueden indicarse los puntos concretos en los que el módulo interactuará con la web. En las versiones anteriores a la 4.7.0 existía una función distinta para crear cada tipo de control que podía utilizarse dentro de cada formulario. Por ejemplo, para los botones existía la función *form_submit*, para las listas de selección *form_select*, y así sucesivamente para cada control (2).

A partir de la versión 4.7.0 todas esas funciones se han eliminado del APIs de Drupal, ahora básicamente todo el trabajo se realiza con una única función llamada *drupal_get_form*. Esta función admite como uno de sus argumentos un array de arrays que contiene todos los controles que quieren añadirse a un formulario, así como sus propiedades concretas (tipo, título, valor por defecto).

Hay que apuntar que las APIs de Drupal parecen cambiar con cada nueva versión y eso podría implicar tener que modificar el código de los módulos.

1.2 Servidor de aplicaciones. Apache.

En informática se denomina servidor de aplicaciones a un servidor en una red de computadores que ejecuta ciertas aplicaciones de software. Usualmente se trata de un dispositivo de software que proporciona servicios de aplicación a las computadoras cliente. Un servidor de aplicaciones generalmente gestiona la mayor parte (o la totalidad) de las funciones de lógica de negocio y de acceso a los datos de la aplicación. Los principales beneficios de la aplicación de la tecnología de servidores de aplicación son la centralización y la disminución de la complejidad en el desarrollo de aplicaciones.

Apache ha demostrado ser sustancialmente más rápido que muchos otros servidores libres y compite de cerca con los mejores servidores comerciales. Es el servidor HTTP más usado de los sitios web.

El servidor Apache es un software que está estructurado en módulos. La configuración de cada módulo se hace mediante la configuración de las directivas que están contenidas dentro del módulo. Los módulos del Apache se pueden clasificar en distintas categorías:

Módulos Base: Módulo con las funciones básicas del Apache.

Módulos Multiproceso: son los responsables de la unión con los puertos de la máquina, aceptando las peticiones y enviando a los hijos a atender a las peticiones.

Las funcionalidades más elementales se encuentran en el módulo base, siendo necesario un módulo multiproceso para manejar las peticiones. Se han diseñado varios módulos multiproceso para cada uno de los sistemas operativos sobre los que se ejecuta el Apache, optimizando el rendimiento y rapidez del código.

El resto de funcionalidades del servidor se consiguen por medio de módulos adicionales que se pueden cargar. Para añadir un conjunto de utilidades al servidor, simplemente hay que añadirle un módulo, de forma que no es necesario volver a instalar el software.

En el fichero se encuentran todos los parámetros de funcionamiento del Apache. Algunos parámetros son generales para la instalación y funcionamiento del Apache. Muchos otros de los parámetros se pueden configurar independientes para un conjunto de directorios y/o ficheros. En estos casos los parámetros se encuentran ubicados dentro de secciones donde se indica el ámbito de aplicación del parámetro.

En primer lugar hay que destacar que el fichero está dividido en tres secciones, que son:

- Parámetros globales.
- Directivas de funcionamiento.
- Host virtuales.

1.3 Metodología: Rational Unified Process (RUP).

El RUP es un proceso de desarrollo de software dirigido por casos de uso, centrado en la arquitectura, iterativo e incremental. RUP pretende implementar las mejores prácticas en ingeniería de software, con el objetivo de asegurar la producción de software de calidad, dentro de plazos y presupuestos predecibles.

Los Casos de Uso son una técnica de captura de requisitos que fuerza a pensar en términos de importancia para el usuario y no sólo en términos de funciones que sería bueno contemplar. En RUP los Casos de Uso no son sólo una herramienta para especificar los requisitos del sistema. También guían su diseño, implementación y prueba. Los Casos de Uso constituyen un elemento integrador y una guía del trabajo.

Los elementos del RUP son:

- **Actividades:** Son los procesos que se llegan a determinar en cada iteración.
- **Trabajadores:** Son las personas o entes involucrados en cada proceso.
- **Artefactos:** Un artefacto puede ser un documento, un modelo, o un elemento de modelo.

1.3.1 Principales características:

1. **Desarrollo iterativo:** Permite una comprensión creciente de los requerimientos, a la vez que se va haciendo crecer el sistema. RUP sigue un modelo iterativo que aborda las tareas más riesgosas

primero. Así se logra reducir los riesgos del proyecto y tener un subsistema ejecutable tempranamente.

2. **Administración de requerimientos:** RUP describe cómo obtener los requerimientos, cómo organizarlos, cómo documentar los requerimientos de funcionalidad y restricciones, cómo rastrear y documentar las decisiones, y cómo captar y comunicar los requerimientos del negocio.
3. **Arquitecturas basadas en componentes:** El proceso se basa en diseñar tempranamente una arquitectura base ejecutable. Esta arquitectura debe ser: flexible, fácil de modificar, intuitivamente comprensible, y debe promover la reutilización de componentes.
4. **Modelamiento visual:** RUP propone un modelamiento visual de la estructura y el comportamiento de la arquitectura y las componentes. En este esquema, los bloques de construcción deben ocultar detalles, permitir la comunicación en el equipo de desarrollo, y permitir analizar la consistencia entre las componentes, entre el diseño y entre la implementación. UML es la base del modelamiento visual de RUP.
5. **Verificación de la calidad del software:** No sólo la funcionalidad es esencial, también el rendimiento y la confiabilidad. RUP ayuda a planificar, diseñar, implementar, ejecutar y evaluar pruebas que verifiquen estas cualidades.
6. **Control de cambios:** Los cambios son inevitables, pero es necesario evaluar si éstos son necesarios y también es necesario rastrear su impacto. RUP indica como controlar, rastrear y monitorear los cambios dentro del proceso iterativo de desarrollo.

RUP divide el proceso de desarrollo en ciclos, donde se obtiene un producto al final de cada ciclo. Cada ciclo se divide en cuatro Fases: Concepción, Elaboración, Construcción, y Transición. Cada fase concluye con un hito bien definido donde deben tomarse ciertas decisiones.

1. Fase de concepción.

En esta fase se establece la oportunidad y alcance el proyecto. Se identifican todas las entidades externas con las que se trata (actores) y se define la interacción en un alto nivel de abstracción: se deben identificar

todos los casos de uso, y se deben describir algunos en detalle. La oportunidad del negocio incluye: definir los criterios de éxito, identificación de riesgos, estimación de recursos necesarios, y plan de las fases incluyendo hitos.

2. Fase de elaboración.

Definir y validar una arquitectura estable. Se hace un refinamiento de la visión del sistema, basándose en nueva información obtenida durante esta fase, se establece una sólida comprensión de los casos de uso más críticos que definen las decisiones arquitectónicas y de planificación. Creación de los planes de desarrollo detallados para las iteraciones de la fase de construcción.

3. Fase de construcción.

Gestión de los recursos, optimización y control de los procesos de construcción del software. Se completa el desarrollo de los componentes y/o subsistemas, probándolos contra un conjunto definido de criterios aprobados al inicio del proyecto.

4. Fase de transición.

Ejecución de los planes de implantación. Se finalizan los manuales de usuario y mantenimiento. Pruebas del sistema en el entorno de explotación. Creación de una release del sistema. Validación del sistema por los usuarios. Ajuste fino del sistema según la validación con el usuario. Se facilita la transición del sistema al personal de mantenimiento. Se pone el producto a disposición del usuario obteniéndose una versión beta del producto.

1.3.2 Lenguaje de Modelaje Unificado (UML).

El UML, es el lenguaje gráfico de modelaje orientado a objetos estándar de la industria para visualizar, construir y documentar los elementos de los sistemas de software. UML proporciona una forma estándar de escribir los planos de un sistema, cubriendo tanto las cosas conceptuales, tales como procesos del negocio y funciones del sistema, como las cosas concretas, tales como las clases escritas en un lenguaje de programación específico, esquemas de bases de datos y componentes de software reutilizables. Simplifica el proceso complejo de análisis y diseño de software, facilitando un plano para la construcción.

El UML representa una colección de las mejores prácticas de ingeniería que han probado ser exitosas en el modelado de sistemas grandes y complejos, define una colección de notaciones para los diferentes diagramas y elementos de modelado que lo componen. Fue diseñado para usarse con un proceso iterativo, incremental, guiado por casos de uso y centrado en la arquitectura.

UML ayuda al usuario a entender la realidad de la tecnología y la posibilidad de que reflexione antes de invertir y gastar grandes cantidades en proyectos que no estén seguros en su desarrollo, reduciendo el coste y el tiempo empleado en la construcción de las piezas que constituirán el modelo.

UML es solo un lenguaje de modelado. No define un proceso que diga cómo utilizar UML para desarrollar sistemas de software.

UML brinda el lenguaje de aplicación de modelado para:

- Modelado de proceso de negocios con casos de uso.
- Modelado de clases y objetos.
- Modelado de componentes.
- Modelado de distribución y despliegues.

Dentro de las partes de UML podemos encontrar:

- **Vistas:** Las vistas muestran diferentes aspectos de los sistemas que son modelados. Una vista no es un gráfico, pero es una abstracción que consiste en una serie de diagramas. Solamente definiendo una serie de vistas, cada una mostrando un aspecto particular del sistema, puede ser construida una imagen completa del sistema. Las vistas también enlazan el lenguaje de modelaje al proceso/método escogido para el desarrollo.
- **Diagramas:** Son los gráficos que describen los contenidos en una vista. El UML tiene nueve tipos diferentes de diagramas que son utilizados en combinación para proporcionar todas las vistas del sistema.

- **Elementos del modelo:** Los conceptos utilizados en los diagramas son los elementos del modelo los cuales representan conceptos orientados a objetos comunes, tales como clases, objetos, mensajes, y las relaciones entre estos conceptos incluyendo asociación, dependencia y generalización. Un elemento del modelo es utilizado en varios diagramas diferentes, pero siempre tiene el mismo significado y símbolo.

En resumen el Lenguaje Unificado de Modelado (UML) es un lenguaje de modelado visual que se usa para especificar, visualizar, construir y documentar artefactos de un sistema de software. Captura decisiones y conocimiento sobre los sistemas que se deben construir. Se usa para entender, diseñar, hojear, configurar, mantener, y controlar la información sobre tales sistemas. UML incluye conceptos semánticos, notación, y principios generales. Tiene partes estáticas, dinámicas, de entorno y organizativas. Está pensado para ser utilizado en herramientas interactivas (Herramientas-Cases) de modelado visual que tengan generadores de código así como generadores de informes.

1.3.3 Herramientas CASE – UML

Las Herramientas CASE (Ingeniería de Software Asistida por Ordenador) son diversas aplicaciones informáticas destinadas a aumentar la productividad en el desarrollo de software reduciendo el coste de las mismas en términos de tiempo y de dinero. Estas herramientas pueden ayudar en todos los aspectos del ciclo de vida de desarrollo del software en tareas como el proceso de realizar un diseño del proyecto, cálculo de costes, implementación de parte del código automáticamente con el diseño dado, compilación automática, documentación o detección de errores entre otras aplicaciones.

Objetivos de las Herramientas Cases:

- Mejorar la productividad en el desarrollo y mantenimiento del software.
- Aumentar la calidad del software.
- Mejorar el tiempo y coste de desarrollo y mantenimiento de los sistemas informáticos.
- Mejorar la planificación de un proyecto.

- Aumentar la biblioteca de conocimiento informático de una empresa ayudando a la búsqueda de soluciones para los requisitos.
- Automatizar, desarrollo del software, documentación, generación de código, pruebas de errores y gestión del proyecto.
- Ayuda a la reutilización del software, portabilidad y estandarización de la documentación.
- Gestión global en todas las fases de desarrollo de software con una misma herramienta.
- Facilitar el uso de las distintas metodologías propias de la ingeniería del software.

En la universidad se utiliza la herramienta propietaria en su versión libre para Linux Visual Paradigm 2.3 la cual es la que se pretende utilizar en la realización de casos de uso, análisis diseño, implementación y prueba para el módulo estadístico.

El Visual Paradigm es una herramienta CASE que utiliza “UML”: como lenguaje de modelaje, con el uso del acercamiento orientado al objeto. Esta herramienta apoya los estándares más altos de las notaciones de Java y de UML. Las transiciones del análisis al diseño y la puesta en práctica se integran dentro de la herramienta. Genera productos de calidad, soporta aplicaciones web y es fácil de instalar y actualizar.

1.3.4 Características del Visual Paradigm:

- Entorno de creación de diagramas para UML 2.0.
- Diseño centrado en casos de uso y enfocado al negocio que generan un software de mayor calidad.
- Uso de un lenguaje estándar común a todo el equipo de desarrollo que facilita la comunicación.
- Capacidades de ingeniería directa (versión profesional) e inversa.
- Modelo y código que permanece sincronizado en todo el ciclo de desarrollo.
- Disponibilidad de múltiples versiones, para cada necesidad.
- Disponibilidad de integrarse en los principales IDEs. (Integrated Development Environment, un entorno de desarrollo integrado que es un programa compuesto por un conjunto de herramientas para un programador).
- Disponibilidad en múltiples plataformas.

Características de los Paquetes de Trabajo.

- Ofrecen un mecanismo general para la organización de los modelos/subsistemas/capas agrupando elementos de modelado.
- Cada paquete se corresponde a un submodelo (subsistema) del modelo (sistema)
- Se pueden anidar paquetes.
- Una clase de un paquete puede aparecer en otro paquete por importación a través de una relación de dependencia entre paquetes.

1.4 PHP

Es un lenguaje de programación gratuito e independiente de plataforma, rápido, con una gran librería de funciones y mucha documentación. Actualmente PHP se encuentra en su versión PHP 5.

PHP puede ser utilizado en cualquiera de los principales sistemas operativos del mercado, incluyendo Linux, muchas variantes Unix (incluido HP-UX, Solaris y OpenBSD), Microsoft Windows, Mac OS X, RISC OS y probablemente alguno más. PHP soporta la mayoría de servidores web de hoy en día, incluyendo Apache, Microsoft Internet Information Server, Personal Web Server, Netscape e iPlanet, PostgreSQL, entre otros. Tiene módulos disponibles para la mayoría de los servidores, para aquellos otros que soporten el estándar CGI, PHP puede usarse como procesador CGI (3).

En el desarrollo de servicios web, PHP no se ha quedado atrás, aunque no poseen una extensión que se encargue de ello. Basándose en SOAP, el protocolo más popular para la creación y consumo de servicios web su principal implementación es NuSOAP. La mejor opción es el uso del PHP 5.00 que desde sus inicios incorpora la extensión php_soap para el manejo del protocolo SOAP, además de ser orientado a objetos, ganando en eficiencia, estabilidad y facilidades de uso.

Por qué utilizar PHP y no otras opciones:

- La principal característica por la cual se utilizó php como lenguaje de programación es precisamente porque Drupal esta totalmente implementado con dicho lenguaje.
- Se pueden hacer grandes cosas con pocas líneas de código. Lo que hace que merezca la pena utilizarlo.
- Viene acompañado por una excelente biblioteca de funciones que permite realizar cualquier labor (acceso a base de datos, encriptación, envío de correo, gestión de un comercio electrónico, XML, creación de PDF ...)
- Esta siendo utilizado con éxito en varios millones de sitios web.
- Es multiplataforma, funciona en todas las plataformas que soporten apache.
- Es software libre. Se puede obtener en la web y su código esta disponible bajo la licencia GPL. Este es uno de los aspectos fundamentales para la elección de este lenguaje.

Las cuatro grandes características: Velocidad, estabilidad, seguridad y simplicidad.

- Velocidad: No solo la velocidad de ejecución, la cual es importante, sino además no crear demoras en la máquina. Por esta razón no debe requerir demasiados recursos de sistema. PHP se integra muy bien junto a otro software, especialmente bajo ambientes Unix, cuando se configura como módulo de Apache, esta listo para ser utilizado.
- Estabilidad: La velocidad no sirve de mucho si el sistema se cae cada cierta cantidad de ejecuciones. Ninguna aplicación es 100% libre de bugs, pero teniendo de respaldo una increíble comunidad de programadores y usuarios es mucho mas difícil para lo bugs sobrevivir. PHP utiliza su propio sistema de administración de recursos y dispone de un sofisticado método de manejo de variables, conformando un sistema robusto y estable.
- Seguridad: El sistema debe poseer protecciones contra ataques. PHP provee diferentes niveles de seguridad, estos pueden ser configurados desde el archivo .ini.
- Simplicidad: Se les debe permitir a los programadores generar código productivamente en el menor tiempo posible. Usuarios con experiencia en C y C++ podrán utilizar PHP rápidamente.

Se hizo uso de la librería GD la cual es una extensión de PHP que permite la creación de imágenes de manera dinámica. Esta librería es gratuita y viene incluida en las versiones 4.3.x y posteriores de PHP.

En la universidad es muy utilizado este lenguaje por su facilidad de uso. Poco a poco este lenguaje se ha ido promocionando para el desarrollo de sitios web. Además un gran por ciento de la institución utiliza los CMS, y casi todos están implementados y soportan este lenguaje, a parte de que hoy en día se ha echo necesario en el país la migración al software libre. Las nuevas versiones ya soportan la programación orientada a objeto así como todos sus paradigmas, su código puede ser embebido dentro del código HTML pudiendo crear objetos dinámicos en tiempo de ejecución.

1.5 HTML

El lenguaje HTML puede ser creado y editado con cualquier editor de textos básico, como puede ser Gedit, el Bloc de Notas de Windows (o Notepad), o cualquier otro editor que admita texto sin formato como GNU Emacs, Microsoft Wordpad, TextPad, Vim etc.

Existen además, otros programas para la realización de sitios web o edición de código HTML, como por ejemplo Microsoft FrontPage, el cual tiene un formato básico parecido al resto de los programas de Office. También existe el famoso software de llamado Dreamweaver, siendo uno de los más utilizados en el ámbito de diseño y programación web. Estos programas se les conoce como editores WYSIWYG o What You See Is What You Get (en español: "lo que ves es lo que obtienes"). Esto significa que son editores los cuales van mostrando el resultado de lo que se está editando en tiempo real a medida que se va desarrollando el documento. Ahora bien, esto no significa una manera distinta de realizar sitios web, sino que una forma un tanto más simple ya que estos programas, además de tener la opción de trabajar con la vista preliminar, tiene su propia sección HTML la cual va generando todo el código a medida que se va trabajando.

Combinar estos dos métodos resulta muy interesante, ya que de alguna manera se ayudan entre sí. Por ejemplo; si estoy editando todo en HTML y de pronto olvido algún código o etiqueta, simplemente me dirijo al editor visual o WYSIWYG y continuo ahí la edición, o viceversa, ya que hay casos en que sale más rápido y fácil escribir directamente el código de alguna característica que se desee adherirle al sitio, que buscar la opción en el programa mismo.

HTML utiliza etiquetas o marcas, que consisten en breves instrucciones de comienzo y final, mediante las cuales se determinan la forma en la que debe aparecer en su navegador el texto, así como también las imágenes y los demás elementos, en la pantalla del ordenador.

1.6 MYSQL

MySQL es un sistema de gestión de base de datos, multihilo y multiusuario con más de seis millones de instalaciones. MySQL AB desarrolla MySQL como software libre en un esquema de licenciamiento dual. Por un lado lo ofrece bajo la GNU GPL, pero, empresas que quieran incorporarlo en productos privativos pueden comprar a la empresa una licencia que les permita ese uso. Está desarrollado en su mayor parte en ANSI C (4).

Existen varias APIs que permiten, a aplicaciones escritas en diversos lenguajes de programación, acceder a las bases de datos MySQL, incluyendo C, C++, C# cada uno de estos utiliza una API específica. También existe un interfaz ODBC, llamado MyODBC que permite a cualquier lenguaje de programación que soporte ODBC comunicarse con las bases de datos MySQL.

MySQL es muy utilizado en aplicaciones web. Su popularidad como aplicación web está muy ligada a PHP, que a menudo aparece en combinación con MySQL. Es una base de datos muy rápida en la lectura cuando utiliza el motor no transaccional MyISAM, pero puede provocar problemas de integridad en entornos de alta concurrencia en la modificación. En aplicaciones web hay baja concurrencia en la modificación de datos y en cambio el entorno es intensivo en lectura de datos, lo que hace a MySQL ideal para este tipo de aplicaciones.

MySQL de forma general es un sistema de administración de bases de datos muy popular en el mundo entero. Una base de datos es una colección estructurada de tablas que contienen datos. Esta puede ser desde una simple lista de compras a una galería de pinturas o el vasto volumen de información en una red corporativa. Para agregar, acceder a y procesar datos guardados en un computador, usted necesita un administrador como MySQL Server. Dado que los computadores son muy buenos manejando grandes cantidades de información, los administradores de bases de datos juegan un papel central en computación, como aplicaciones independientes o como parte de otras aplicaciones.

Además MySQL es un sistema de administración relacional de bases de datos. Una base de datos relacional archiva datos en tablas separadas en vez de colocar todos los datos en un gran archivo. Esto

permite velocidad y flexibilidad. Las tablas están conectadas por relaciones definidas que hacen posible combinar datos de diferentes tablas.

En estos momentos el país está en proceso de migración al software libre. MYSQL es software de fuente abierta. Fuente abierta significa que es posible para cualquier persona usarlo y modificarlo. Cualquier interesado puede estudiar el código fuente y ajustarlo a sus necesidades. MySQL usa el GPL (GNU General Public License) para definir que puede hacer y que no puede hacer con el software en diferentes situaciones. Si usted no se ajusta al GPL o requiere introducir código MySQL en aplicaciones comerciales, usted puede comprar una versión comercial licenciada.

- Según las cifras del fabricante, existirían más de seis millones de copias de MySQL funcionando en la actualidad, lo que supera la base instalada de cualquier otra herramienta de bases de datos.

En este capítulo se hizo un análisis del estudio del arte de las principales tecnologías y herramientas que se utilizaron en la realización del sistema y se definieron las tecnologías y herramientas a utilizar en el desarrollo de aplicaciones web. A partir de este análisis se llegó a las siguientes conclusiones.

- Se instalaron todos los servicios y herramientas libres de desarrollo de Drupal especialmente PHP como lenguaje de programación el cual ofrece una gran productividad y componente ineludible para la entrega de la aplicación.
- Se hizo un análisis del funcionamiento de las APIs de Drupal.

Capítulo 2. Características del sistema.

En este capítulo se comenzará por la descripción de las principales características que dieron origen al sistema estadístico de Drupal para el proyecto, primeramente se realizará un análisis de los objetivos estratégicos del proyecto y procesos que los soportan, así como un estudio crítico de los procesos que se ejecutan actualmente y cuales fueron las causas que originaron el desarrollo del mismo. Una vez comprendido el problema, se pasará a explicar los procesos que serán objeto de automatización así como la relación que pudiera tener con algún otro sistema automatizado de la institución. Luego se hará referencia a los documentos específicos que sirvieron de apoyo para el desarrollo del sistema. Terminado el entendimiento del problema planteado, se iniciará la descripción general de la solución propuesta, realizando un análisis comparativo con otras propuestas existentes, llegando a conclusiones de las ventajas o desventajas de la propuesta. Luego se entrará a indicar como se solucionar los problemas que dieron origen al sistema, para una mayor comprensión se hará una representación de los casos de uso del sistema, indicando quienes son los actores. Todo lo antes mencionado da origen a los requisitos funcionales, en los cuales se recogen las características básicas que tendrá el sistema, además de indicar los requisitos no funcionales que son necesarios para garantizar el éxito del sistema estadístico. Concluirá este capítulo con la descripción detallada de cada uno de los casos de uso del sistema, los cuales guiarán las siguientes etapas del desarrollo del software.

2.1 Problema.

Actualmente el proceso estadístico se lleva a cabo en la mayoría de los sitios Web del mundo, un gran % de los sitios web llevan un estricto control de las informaciones estadísticas del sitio. Muchos de las web están montadas sobre CMS ejemplo de estos Drupal que brinda una serie de funcionalidades que hace que los sitios sean usables y accesibles. Recientes encuestas han demostrado que un control efectivo de la estadística de un sitio brinda una mayor funcionalidad de la web. Los fallos con los controles estadísticos pueden contribuir a un uso inadecuado del sitio. Con el objetivo de dar respuesta a este problema, se ha dado la tarea de desarrollar el sistema estadístico que cumpla con las necesidades

específicas del proyecto, en el cual se recojan los procesos estadísticos de la web, lo cual posibilitaría que el administrador del sitio pueda conocer todos los movimientos de un usuario en el sitio.

2.2 Objeto de Automatización

En busca de una mayor funcionalidad del sistema se llevaron a cabo una serie de procesos que serán objeto de automatización y se llegó a la conclusión de organizar todos los movimientos estadísticos por usuario, las páginas visitadas, y las operaciones realizadas dentro de esta, las estadísticas de cada uno de los accesos hechos al sitio y la representación gráfica de cada proceso estadístico ya sea organizado por usuario o por páginas visitadas en distintos modelos gráficos.

2.3 Información que se maneja

Para el desarrollo del sistema se ha tomado como base el módulo estadístico de la comunidad de Drupal, creado por sus desarrolladores hace algunos años el cual constituye una versión anterior al implementado en el proyecto. Se partió de los criterios mas actualizados recogidos en el centro. También se incluyó en el sistema los criterios de los desarrolladores de módulos en los diferentes proyectos en la Universidad los cuales contribuyeron en gran medida en el conocimiento implícito del módulo, además de la vinculación del mismo con todos los sitios web que de una manera u otra llevan el control estadístico de su sitio en cuestión.

2.4 Propuesta del sistema

En relación con las necesidades del módulo se llevaron a cabo varias propuestas del sistema de las cuales se decidió hacer una comparación de las mismas con el objetivo de extraer la más factible. Como primera propuesta fue la representación gráfica de los movimientos en la web representados por diferentes criterios. El sistema también estará dirigido en inicio para el proyecto productivo, pero puede ser utilizado por todos aquellos que de una manera u otra utilizan este CMS la cual podrá adquirir en el mismo un conjunto de medidas y consejos que pueden ayudar a mejorar el funcionamiento de su web.

2.5 Modelo del Dominio

Definición de las entidades y los conceptos principales.

Concepto	Descripción
Sistema	Aplicación de software en funcionamiento, accesible al personal de las dependencias de la entidad.
Módulo	Componentes de la aplicación de software en el caso del CMS. Los sistemas están divididos en módulos o subsistemas.
Usuario	Representa a personas o sistemas de la entidad que interactúan con el sitio en Drupal. Inicia todos los procesos de control estadístico.
Rol	Papel que juegan los usuarios dentro del sistema.
Administrador	Representa a personas o sistemas de la entidad que interactúan con el sitio Drupal y pueden editar las informaciones del proceso estadístico. Visualiza y controla la información estadística.
Acción	Acciones que realizan los usuarios sobre los sistemas, ya sean de acceso al sistema y sus módulos, de acceso a datos, consultas, etc.
Evento	Se registra cada vez que un usuario realiza una acción sobre el sistema, es la manera de ver las acciones de la parte del sistema.
BitácoraBd Tabla base datos	Es la estructura donde se almacenan los eventos que ocurren, así con la información específica de la hora, fecha, usuario, páginas visitadas, la acción realizada y otros datos de interés. Es la base de la auditoría.
Grupo de Usuarios	Los grupos de usuarios se conforman por los usuarios que tienen permisos de acceso idéntico. Los grupos de usuario agrupan funcionalidades comunes.

2.6 Representación del modelo del dominio.

El modelo del dominio se hace cuando no es posible llevar a cabo un negocio, este viene siendo un subconjunto de procesos del sistema.

Ver Modelo de Dominio en el Anexo 1.

2.7 Requisitos

Requisitos funcionales.

- RF1 Mostrar las gráficas de visitas al sitio.
- RF2 Visualizar las páginas visitadas, así como agruparlas por IP y por usuario.
- RF3 Almacenar la estadística detallada por usuario y sistema.
- RF4 Visualizar y escoger opciones de configuración del módulo en un intervalo de tiempo.
- RF5 Procesar datos estadísticos.
- RF6 Activar módulo estadístico.

Requisitos no funcionales

Soporte

- El sistema debe correr sobre el gestor de base datos MySQL y PostgreSQL.
- Fácil instalación.

Navegación.

- Según la página del sitio en que se encuentre el usuario habrá un identificador que le indique donde se encuentra.

Software.

- Se necesita un ordenador que tenga entre sus programas MySQL en su versión 5.0 y PHP.

Portabilidad

Sistemas operativos.

- Microsoft Windows 98, NT, 2000, XP o Vista.
- GNU/ Linux.

Ayuda y documentación en línea.

- El módulo cuenta con un sistema de ayuda en línea que orienta al usuario sobre las posibilidades que brinda el mismo.

Usabilidad

La interfaz de usuario del sistema es amigable y fácilmente comprendida permitiendo al administrador una interacción efectiva con el sistema, además cumple con los requisitos siguientes:

- Las ventanas del sistema contienen claro y bien estructurado los datos, y al mismo tiempo permiten la interpretación correcta e inequívoca de la información.
- Todos los textos y mensajes en pantalla aparecen en idioma castellano.
- Fácil para ser comprendido.

Seguridad

- Esta implementado a través de varios niveles de usuarios, con permisos que correspondan al rol que desempeñan en la aplicación ya que Drupal cuenta con un módulo users el cual verifica los permisos de cada usuario.

2.8 Casos de Uso del sistema.

El modelo de casos de uso describe lo que hace el sistema para cada tipo de usuario. Cada uno de estos se representa mediante uno o más actores.

Un actor del sistema es un rol que desempeña un usuario o sistema externo al interactuar con el sistema que se modela.

2.8.1 Determinación y justificación de los actores del sistema.

Actor	Justificación
Usuario	Es el que interactúa con el sistema, la persona que le da funcionalidad al software.
Administrador	Es la persona que visualiza todos los registros estadísticos del sistema de software.

2.8.2 Descripción y expansión de los casos de uso.

CU-1	Procesar Request
Actor	Usuario
Descripción	Procesa todos los movimientos del usuario al instante en que este accede a la URL del sitio.
Referencia	RF5

CU-2	Visualizar gráficas estadísticas
Actor	administrador
Descripción	Representa las gráficas de visitas al sitio.
Referencia	RF1

CU-3	Visualizar estadísticas de accesos.
Actor	administrador
Descripción	Visualizar toda la información con respecto a los usuarios, páginas e IP que visitan el sitio
Referencia	RF2

CU-4	Visualizar estadística detallada
Actor	administrador
Descripción	Visualiza la información general con respecto a usuarios y el sistema en su totalidad.
Referencia	RF3

CU-5	Escoger opciones estadísticas
-------------	-------------------------------

Actor	administrador
Descripción	Permite al administrador activar o desactivar los registros estadísticos en un intervalo de tiempo escogido por el mismo.
Referencia	RF5

CU-6	Activar módulo estadístico
Actor	Administrador
Descripción	Activar el módulo estadístico para el proceso de recopilación de información
Referencia	RF6

2.9 Diagramas de caso de uso del sistema.

Se utilizó una generalización/especialización entre los actores del sistema debido a que el administrador realiza las mismas operaciones que el usuario dentro del sitio. Además se adicionó a los casos de uso aquel que estaba vinculado con la activación del módulo debido a que era imprescindible la activación del mismo para su funcionamiento. Cada caso de uso responde a una serie de acciones que se ejecutan y que brindan un resultado de valor para un actor y sirvieron para especificar la funcionalidad y el comportamiento de un sistema mediante su interacción con los usuarios y/u otros sistemas.

El diagrama de Casos de Uso del sistema puede verse en el Anexo 2.

2.10 Casos de Uso expandidos.

A través de la expansión de los casos de uso se describe paso a paso la secuencia de eventos que los actores utilizan para completar un proceso a través del sistema.

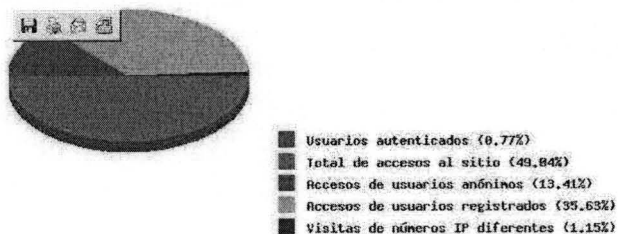
Caso de uso	
CU-1	Procesar Request.
Propósito	Ejecutar todos los movimientos estadísticos por el sistema que se realicen en el sitio.

Actores	usuario
Resumen:	El caso de uso se inicia cuando el usuario accede a cualquier URL definida en el sitio.
Referencias	RF5
Precondiciones	
Poscondiciones	
Acción del actor	Respuesta del sistema
1. El usuario accede a una URL del sitio.	1.1 El sistema empieza a procesar y almacenar todos los movimientos del usuario dentro del sitio.
2. El usuario se desconecta.	2.1 El sistema queda inhabilitado.
Flujo alternativo de los eventos	
Acción del actor	Respuesta del sistema
Puntos de extensión.	

Caso de uso	
CU-2	Visualizar gráficas estadísticas.
Propósito	Procesar las gráficas estadísticas.
Actores	administrador
Resumen:	El caso de uso se inicia cuando el administrador accede directamente a la URL o interfaz definida.
Referencias	RF1
Precondiciones	Las gráficas existan y el usuario tenga los permisos de acceso.
Poscondiciones	

- admin
 - crear contenido nuevo
 - mi cuenta
 - ▼ administrar
 - bloques
 - categorías
 - comentarios
 - contenido
 - control de acceso
 - formatos de entrada
 - image publishing
 - localización
 - módulos
 - menús
 - opciones
 - ▼ registros
 - accesos recientes
 - páginas visitadas
 - principales visitantes
 - ▼ estadísticas
 - detalladas
 - representación gráfica
 - temas
 - usuarios

Representación de los datos de usuarios. Gráfica de pastel



Acción del actor	Respuesta del sistema
1. El administrador se autentica en el sistema.	1.1 El módulo encargado comprueba que el usuario tenga privilegios suficientes para visualizar dicho contenido.
2. El administrador accede a la sección administrar	2.1 El sistema muestra las opciones de administración.
3. El administrador accede a la pestaña de registros (logs).	3.1 El sistema muestra todos los registros estadísticos que se procesan
4. El administrador accede a la pestaña estadísticas detalladas.	4.1 El sistema despliega la pestaña de representación gráfica.
5. . El administrador accede a la pestaña representación gráfica.	5.1 El módulo carga desde la base de datos el contenido lo procesa y muestra las gráficas correspondientes.

6 .El administrador visualiza la información.	
Flujo alternativo (8)	
Acción del actor	Respuesta del sistema
	1.1 No tiene permisos de administración
Puntos de extensión.	

Caso de uso	
CU-3	Visualizar estadísticas de acceso.
Propósito	Controlar todos los accesos de usuarios agrupados por páginas.
Actores	administrador
Resumen: El caso de uso inicia cuando el administrador accede directamente a la URL del registro estadístico de usuarios y el sistema redirecciona al usuario de una sección a la URL del contenido. El sistema comprueba que el usuario tenga privilegios suficientes para visualizar dicho contenido y en caso de tenerlos carga desde la base de datos el contenido y lo muestra en formato HTML	
Referencias	RF2
Precondiciones	El usuario haya accedido a la URL del sitio.
Poscondiciones	
Acción del actor	Respuesta del sistema
1. El administrador accede directamente a la URL del contenido.	<p>1.1 El módulo encargado comprueba que el usuario tenga privilegios suficientes para visualizar dicho contenido.</p> <p>1.2 El sistema determina que tipo de contenido se esta solicitando y delega la responsabilidad al módulo que gestione dicho tipo de contenido.</p> <p>1.3 EL sistema carga de la base de datos la información gestionada.</p>

<p>Sección “Accesos de IP”</p>	<p>admin</p> <ul style="list-style-type: none"> ▷ crear contenido nuevo ▷ mi cuenta ▷ administrar <ul style="list-style-type: none"> ▷ bloques ▷ categorías ▷ comentarios ▷ contenido ▷ control de acceso ▷ formatos de entrada ▷ image publishing ▷ localización ▷ módulos ▷ menús ▷ opciones ▷ registros <ul style="list-style-type: none"> ▷ accesos recientes ▷ Páginas visitadas ▷ Principales visitantes ▷ estadísticas detalladas 	<table border="1"> <thead> <tr> <th>accesos▼</th> <th>Visitante</th> <th>Tiempo total de generación de páginas</th> </tr> </thead> <tbody> <tr> <td>5</td> <td>10.8.126.204</td> <td>1 seg</td> </tr> <tr> <td>4</td> <td>admin</td> <td>3 segs</td> </tr> <tr> <td>2</td> <td>yeisel</td> <td>1 seg</td> </tr> <tr> <td>1</td> <td>10.33.14.128</td> <td>2 segs</td> </tr> </tbody> </table>	accesos▼	Visitante	Tiempo total de generación de páginas	5	10.8.126.204	1 seg	4	admin	3 segs	2	yeisel	1 seg	1	10.33.14.128	2 segs
accesos▼	Visitante	Tiempo total de generación de páginas															
5	10.8.126.204	1 seg															
4	admin	3 segs															
2	yeisel	1 seg															
1	10.33.14.128	2 segs															
<p>2 .El administrador accede a la URL del contenido de IP.</p>	<p>2.1 El módulo carga desde la base de datos el contenido lo procesa y lo muestra al sistema en formato HTML.</p> <p>2.2 El sistema muestra el resultado al administrador.</p>																
<p>3 .El administrador visualiza el contenido</p>																	

Sección “Accesos de usuarios”

admin

- ▾ crear contenido nuevo
- ▾ mi cuenta
- ▾ administrar
 - ▾ bloques
 - ▾ categorías
 - ▾ comentarios
 - ▾ contenido
 - ▾ control de acceso
 - ▾ formatos de entrada
 - ▾ image publishing
 - ▾ localización
 - ▾ módulos
 - ▾ menús
 - ▾ opciones
 - ▾ registros
 - ▾ accesos recientes
 - ▾ Páginas visitadas
 - ▾ Principales visitantes

accesos recientes

Fecha	Página	Usuario	Navegador	Sistema Opera
2007-05-30 22:07	administar admin	admin	Internet Explorer	Windows XP
2007-05-30 22:07	node	admin	Internet Explorer	Windows XP
2007-05-30 22:07	node	admin	Internet Explorer	Windows XP
2007-05-30 22:07	node	Anonymous	Internet Explorer	Windows XP
2007-05-30 22:07	log out logout	Anonymous	Internet Explorer	Windows XP
2007-05-30 22:07	posts admin/settings/node	yeisel	Internet Explorer	Windows XP
2007-05-30 22:07	settings admin/settings	yeisel	Internet Explorer	Windows XP

2. El administrador accede a la URL del contenido correspondiente a los accesos de usuarios.

2.1 El módulo carga desde la base de datos el contenido lo procesa y lo muestra al sistema en formato HTML.

2.2 El sistema muestra el resultado al administrador.

Sección "Accesos a páginas"		Visitas	Páginas	Promedio de tiempo de generación de la página	Tiempo total de visita página
<ul style="list-style-type: none"> ○ crear contenido nuevo ○ mi cuenta ○ administrar <ul style="list-style-type: none"> ○ bloques ○ categorías ○ comentarios ○ contenido ○ control de acceso ○ formatos de entrada ○ image publishing ○ localización ○ módulos ○ menús ○ opciones ○ registros <ul style="list-style-type: none"> ○ accesos recientes ○ Páginas visitadas ○ Principales visitantes ○ estadísticas detalladas ○ temas 	41	node	789 ms	32 segs	
	39	accesos recientes admin/logs/accesos_recientes	245 ms	8 segs	
	25	estadísticas detalladas admin/logs/estadisticas_detalleadas/...	337 ms	8 segs	
	13	estadísticas detalladas admin/logs/estadisticas_detalleadas	251 ms	3 segs	
	12	registros admin/logs	282 ms	3 segs	
	10	administrar admin	590 ms	6 segs	
	10	estadísticas detalladas admin/logs/estadisticas_detalleadas/...	364 ms	4 segs	
	9	módulos admin/modules	1509 ms	14 segs	
	7	representación gráfica admin/logs/estadisticas_detalleadas/...	444 ms	3 segs	

<p>2. El administrador accede a la URL del contenido referente a las estadísticas de las páginas.</p>	<p>2.1 El módulo carga desde la base de datos el contenido lo procesa y lo muestra al sistema en formato HTML.</p> <p>2.2 El sistema muestra el resultado al administrador.</p>
<p>2 .El administrador visualiza el contenido</p>	
<p>Flujo alternativo</p>	
<p>Acción del actor</p>	<p>Respuesta del sistema</p>
	<p>1.1 En caso de no tener privilegios suficientes el sistema no muestra el menú de registros estadísticos.</p>
<p>Puntos de extensión.</p>	

Caso de uso																																		
CU-4	Visualizar estadística detallada																																	
Propósito	Controlar todos los movimientos que se realizan sobre el sistema y los usuarios de forma general																																	
Actores	administrador																																	
Resumen:	El caso de uso se inicia en el momento que el usuario accede a la URL del contenido. El sistema comprueba que el usuario tenga privilegios suficientes para visualizar dicho contenido y en caso de tenerlos carga desde la base de datos el contenido y lo muestra en formato HTML																																	
Referencias	RF3																																	
Precondiciones	Exista la información																																	
Poscondiciones																																		
Acción del actor	Respuesta del sistema																																	
1. El administrador accede directamente a la URL del contenido.	1.1 El sistema determina que tipo de información se esta solicitando y delega la responsabilidad al módulo que gestione dicho contenido.																																	
Sección "datos usuarios"	<div style="display: flex; justify-content: space-between;"> datos del sistema datos de usuarios </div> <table border="1"> <thead> <tr> <th colspan="2">Datos relacionados con los usuarios</th> <th>Valor</th> </tr> </thead> <tbody> <tr> <td>Usuarios autenticados</td> <td></td> <td>2</td> </tr> <tr> <td>Usuario registrado más activo</td> <td></td> <td>yeisel</td> </tr> <tr> <td>Total de accesos al sitio</td> <td></td> <td>37</td> </tr> <tr> <td>Último usuario registrado</td> <td></td> <td>yeisel</td> </tr> <tr> <td>Accesos de usuarios anónimos</td> <td></td> <td>19</td> </tr> <tr> <td>Accesos de usuarios registrados</td> <td></td> <td>18</td> </tr> <tr> <td>Visitas de números IP diferentes</td> <td></td> <td>5</td> </tr> <tr> <td>Número IP de donde más visitan</td> <td></td> <td>10.8.126.204</td> </tr> <tr> <td>Total de accesos al sitio en 1 hora</td> <td></td> <td>0</td> </tr> <tr> <td>Accesos de usuarios anónimos en 1 hora</td> <td></td> <td>0</td> </tr> </tbody> </table>	Datos relacionados con los usuarios		Valor	Usuarios autenticados		2	Usuario registrado más activo		yeisel	Total de accesos al sitio		37	Último usuario registrado		yeisel	Accesos de usuarios anónimos		19	Accesos de usuarios registrados		18	Visitas de números IP diferentes		5	Número IP de donde más visitan		10.8.126.204	Total de accesos al sitio en 1 hora		0	Accesos de usuarios anónimos en 1 hora		0
Datos relacionados con los usuarios		Valor																																
Usuarios autenticados		2																																
Usuario registrado más activo		yeisel																																
Total de accesos al sitio		37																																
Último usuario registrado		yeisel																																
Accesos de usuarios anónimos		19																																
Accesos de usuarios registrados		18																																
Visitas de números IP diferentes		5																																
Número IP de donde más visitan		10.8.126.204																																
Total de accesos al sitio en 1 hora		0																																
Accesos de usuarios anónimos en 1 hora		0																																

<p>2 .El administrador selecciona la pestaña para ver los datos de los usuarios.</p>	<p>2.2 Es sistema carga de la base de datos la información que es gestionada de forma común y lo muestra al sistema en formato HTML.</p>	
<p>3. El administrador visualiza la información.</p>		
<p>Sección "datos sistema"</p>		
<p>2. El administrador selecciona la pestaña para ver los datos de los usuarios.</p>	<p>2.1 El sistema determina que tipo de contenido se esta solicitando y delega la responsabilidad al módulo que gestione dicho contenido.</p> <p>2.2 Es sistema carga de la base de datos la información que es gestionada de forma común.</p>	
<p>3. El administrador visualiza la información.</p>		
<p>Flujo alternativo</p>		
<p>Acción del actor</p>	<p>Respuesta del sistema</p>	
<p>Puntos de extensión.</p>		

Caso de uso	
CU-5	Escoger opciones estadísticas
Propósito	Seleccionar opciones de configuración al sitio para procesar algunos datos estadísticos y habilitar el funcionamiento del módulo.
Actores	administrador
Resumen: El caso de uso inicia cuando el administrador del sistema accede a la URL con el objetivo de escoger y activar opciones estadísticas.	
Referencias	RF4
Precondiciones	Exista el menú de configuración.
Poscondiciones	Sean guarden los cambios.
Acción del actor	Respuesta del sistema
1. El administrador accede a la URL donde se activa y escogen las opciones de configuración del módulo.	<p>1.1 El módulo encargado comprueba que el usuario tenga privilegios suficientes para visualizar dicho contenido.</p> <p>1.2 El sistema muestra la interfaz de activación del módulo brindando opciones de configuración.</p>
2. El administrador pulsa sobre el botón guardar configuración.	2.1 El sistema guarda los datos de configuración.
Flujo alternativo	
Acción del actor	Respuesta del sistema
	1.1 No muestra los servicios de administración.
Puntos de extensión.	

Caso de uso																													
CU-6	Activar módulo estadístico																												
Propósito	Habilitar el módulo estadístico para comenzar la recopilación de la información																												
Actores	Administrador																												
Resumen:	El caso de uso se inicia cuando el administrador accede a la URL definida en el sitio.																												
Referencias	RF6																												
Precondiciones	Que esté disponible el módulo estadístico para activarlo																												
Poscondiciones	<table border="0"> <tr> <td>poll</td> <td>de selección múltiple.</td> <td></td> <td></td> </tr> <tr> <td>profile</td> <td>Soporte para perfiles de usuario configurables.</td> <td><input type="checkbox"/></td> <td><input type="checkbox"/></td> </tr> <tr> <td>search</td> <td>Permite la búsqueda de palabras por todo el sitio.</td> <td><input type="checkbox"/></td> <td><input type="checkbox"/></td> </tr> <tr> <td>statistics</td> <td>Módulo para el trabajo con estadísticas.</td> <td><input checked="" type="checkbox"/></td> <td><input type="checkbox"/></td> </tr> <tr> <td>story</td> <td>Permite a los usuarios enviar artículos, historias o contenido similar.</td> <td><input checked="" type="checkbox"/></td> <td><input type="checkbox"/></td> </tr> <tr> <td>system</td> <td>Gestión de la configuración general del sitio para administradores.</td> <td><input type="checkbox"/></td> <td><input type="checkbox"/></td> </tr> <tr> <td>taxonomy</td> <td>Activa la categorización del contenido.</td> <td><input checked="" type="checkbox"/></td> <td><input type="checkbox"/></td> </tr> </table>	poll	de selección múltiple.			profile	Soporte para perfiles de usuario configurables.	<input type="checkbox"/>	<input type="checkbox"/>	search	Permite la búsqueda de palabras por todo el sitio.	<input type="checkbox"/>	<input type="checkbox"/>	statistics	Módulo para el trabajo con estadísticas.	<input checked="" type="checkbox"/>	<input type="checkbox"/>	story	Permite a los usuarios enviar artículos, historias o contenido similar.	<input checked="" type="checkbox"/>	<input type="checkbox"/>	system	Gestión de la configuración general del sitio para administradores.	<input type="checkbox"/>	<input type="checkbox"/>	taxonomy	Activa la categorización del contenido.	<input checked="" type="checkbox"/>	<input type="checkbox"/>
poll	de selección múltiple.																												
profile	Soporte para perfiles de usuario configurables.	<input type="checkbox"/>	<input type="checkbox"/>																										
search	Permite la búsqueda de palabras por todo el sitio.	<input type="checkbox"/>	<input type="checkbox"/>																										
statistics	Módulo para el trabajo con estadísticas.	<input checked="" type="checkbox"/>	<input type="checkbox"/>																										
story	Permite a los usuarios enviar artículos, historias o contenido similar.	<input checked="" type="checkbox"/>	<input type="checkbox"/>																										
system	Gestión de la configuración general del sitio para administradores.	<input type="checkbox"/>	<input type="checkbox"/>																										
taxonomy	Activa la categorización del contenido.	<input checked="" type="checkbox"/>	<input type="checkbox"/>																										
Acción del actor	Respuesta del sistema																												
1. El administrador accede a la URL del sitio(?q=admin/modules)	<p>1.1 El módulo encargado comprueba que el usuario tenga privilegios suficientes para visualizar dicho contenido.</p> <p>1.2 El sistema muestra la interfaz de activación del módulo estadístico.</p>																												
2. El administrador busca en la lista de módulos y selecciona la pestaña de activación(Activado) para el módulo estadístico	2.1 El sistema guarda los cambios y lo carga en la base datos.																												
Flujo alternativo de los eventos																													
Acción del actor	Respuesta del sistema																												
	El sistema muestra un mensaje de error en caso de que el módulo presente algún problema. Volver al flujo básico 2.																												
Puntos de extensión.																													

En este capítulo se llevaron a cabo una serie de tareas de vital importancia a la hora de realizar un módulo estadístico. Se llegó a la conclusión que no hizo falta realizar un modelo del negocio y si, un modelo de dominio ya que no se tenía un cliente directo. Se diseñó el dominio de la aplicación así como las interacciones existentes entre cada objeto. Se extrajeron los requisitos funcionales y no funcionales de la aplicación. Se hizo un estudio de las capacidades y funcionalidades que el sistema debía cumplir así como el levantamiento de los requisitos que hacían al sistema estadístico usable, rápido y confiable. Luego se pasó a la selección de los casos de uso partir de los requerimientos funcionales y los actores que participaban es su ejecución para llegar así al diagrama de casos de uso del sistema donde se visualiza la interacción de cada actor con los casos de uso del sistema a automatizar. Además se describieron los casos de uso para así controlar las secuencias de acciones a partir de la interacción actor y caso de uso sirviendo de base para llevar a cabo las siguientes fases del proyecto.

Capítulo 3. Análisis y diseño del sistema.

El presente capítulo tiene como objetivo llevar a cabo la fase de análisis y diseño del sistema como parte del desarrollo del software cuyo propósito primario es formular el modelo conceptual y de diseño del problema. El diseño es la etapa del proceso de desarrollo de donde se decide como se llevará a cabo el sistema. A través de esta fase se toman decisiones estratégicas y tácticas para cumplir los requerimientos funcionales y de calidad de un sistema.

3.1 Análisis del sistema

3.1.1 Modelo conceptual de las clases de análisis.

El modelo conceptual se realizó un diagrama de clases en notación UML de forma simplificada. Se utilizan las clases preliminares, las asociaciones preliminares y los nombres para las clases.

El modelo conceptual de las clases de análisis se muestra en el Anexo 3

3.2 Análisis del diseño.

En el presente epígrafe se plasman los resultados de la etapa de diseño del sistema utilizando UML. Se definen los diagramas de interacción a partir de las operaciones de los diagramas de colaboración.

3.2.1 Diagramas de interacción.

Los diagramas de interacción no son más que una descripción del modo en que cada operación detectada en los diagramas de colaboración lleva a cabo sus responsabilidades y modifica el estado del sistema, los mismos pueden representarse a través de los Diagramas de Colaboración y/o los Diagramas de Secuencia.

El tipo de diagrama seleccionado para construir los diagramas de interacción fue el de colaboración, donde se describe la comunicación de las clases a través de mensajes.

En el Anexo 4 se muestra el modelo de Análisis por caso de uso.

3.2.2 Descripción de las clases de análisis.

Clases controladoras

Nombre: CC_búsqueda	
Tipo de clase: Controladora	
Atributo	Tipo
Para cada operación:	
Nombre:	statistics_visitantes ()
Descripción:	Muestra todos los datos de los usuarios.

Nombre: CC_gestor_solicitud_registros.	
Tipo de clase: Controladora	
Atributo	Tipo
Para cada operación:	
Nombre:	statistics_visitantes ()
Descripción:	Muestra todos los datos referentes a las páginas del sitio de Drupal.

Nombre: CC_gestor_request	
Tipo de clase: Controladora	
Atributo	Tipo
Para cada operación:	
Nombre:	
Descripción:	Gestiona todos los movimientos de los usuarios dentro del sitio.

Nombre: CC_opciones_estadísticas.	
Tipo de clase: Controladora	
Atributo	Tipo
Para cada operación:	
Nombre:	statistics_seetings ()
Descripción:	Permite activar las opciones del módulo para que comience a procesar la información.

Nombre: CC_gestor_activación	
Tipo de clase: Controladora	
Atributo	Tipo
Para cada operación:	
Nombre:	
Descripción:	Permite que el módulo quede activado para que comience a procesar la información.

Nombre: CC_datos_usuario	
Tipo de clase: Controladora	
Atributo	Tipo
Para cada operación:	
Nombre:	
Descripción:	Procesa datos de los usuarios.

Clases interfaz

Nombre: CI_configurar	
Tipo de clase: Interfaz	
Atributo	Tipo
opciones de registro de acceso enable disable opciones	fieldset radio radio select
Para cada operación:	
Nombre:	CI_configurar
Descripción:	Configurar las opciones de registro estadístico.

Nombre: CI_módulos	
Tipo de clase: Interfaz	
Atributo	Tipo
Para cada operación:	
Nombre:	CI_módulos
Descripción:	Muestra los módulos disponibles en el sistema.

Nombre: CI_gestionar	
Tipo de clase: Interfaz	
Atributo	Tipo
Para cada operación:	
Nombre:	CI_gestionar
Descripción:	Procesa todos los datos estadísticos cuando el usuario está visitando el sitio.

Nombre: CI_Ver gráficas	
Tipo de clase: Interfaz	
Atributo	Tipo
Para cada operación:	
Nombre:	CI_Ver gráficas
Descripción:	Solicita la representación gráfica.

Nombre: CI_ acceso páginas	
Tipo de clase: Interfaz	
Atributo	Tipo
Para cada operación:	
Nombre:	CI_ acceso páginas
Descripción:	Solicita los datos de accesos a páginas.

Nombre: CI_datos sistema	
Tipo de clase: Interfaz	
Atributo	Tipo
Para cada operación:	
Nombre:	CI_datos sistema
Descripción:	Solicita los datos del sistema en cuanto a nodos, comentarios etc.

Nombre: CI_accesos usuario	
Tipo de clase: Interfaz	
Atributo	Tipo
Para cada operación:	
Nombre:	CI_accesos usuario
Descripción:	Solicita el nombre del usuario, hora, navegador... de acceso a cualquier página.

Nombre: CI_datos usuario	
Tipo de clase: Interfaz	
Atributo	Tipo
Para cada operación:	
Nombre:	CI_datos usuario
Descripción:	Solicita los datos del usuario correspondiente al sitio.

Nombre: CI_accesos IP	
Tipo de clase: Interfaz	
Atributo	Tipo
Para cada operación:	
Nombre:	CI_accesos IP
Descripción:	Solicita los datos de los IP que acceden al sitio.

Clases Entidad

Nombre: CE_users	
Tipo de clase: Entidad	
Atributo	Tipo
uid	int
name	varchar
pass	varchar
mail	varchar
mode	tinyint
threshold	tinyint
theme	tinyint

signature	varchar
created	varchar
access	int
login	int
status	int
timezone	tinyint
language	varchar
picture	varchar
init	varchar
data	longtext
Para cada operación:	
Nombre:	CE_users
Descripción:	Almacena datos de los usuarios.

Nombre: CE_accesslog	
Tipo de clase: Entidad	
Atributo	Tipo
aid	int
sid	varchar
title	varchar
path	varchar
url	varchar
hostname	varchar
uid	int
timer	int
timestamp	int
navegador	varchar
sistema_operativo	varchar
Para cada operación:	
Nombre:	CE_accesslog
Descripción:	Almacena datos de los accesos a páginas.

Nombre: CE_system	
Tipo de clase: Entidad	
Atributo	Tipo
filename	varchar
name	varchar
type	varchar
description	varchar
status	int
throttle	tinyint
booststrap	int
shema_version	smallint
weight	int
Para cada operación:	
Nombre:	CE_system
Descripción:	Almacena datos del sistema, módulos .etc.

Nombre: CE_node	
Tipo de clase: Entidad	
Atributo	Tipo
nid	int
vid	int
type	varchar
title	varchar
uid	int
status	int
created	int
changed	int
comment	int
promote	int
moderate	int
sticky	int
Para cada operación:	
Nombre:	CE_node
Descripción:	Almacena los datos de las operaciones sobre los nodos.

Nombre: CE_comments	
Tipo de clase: Entidad	
Atributo	Tipo
cid	int
pid	int
nid	int
uid	int
subject	varchar
comment	longtext
hostname	varchar
timestamp	int
score	mediumint
status	tinyint
format	int
thread	varchar
user	longtext
name	varchar
mail	varchar

homepage	varchar
Para cada operación:	
Nombre:	CE_comments
Descripción:	Almacena datos de los comentarios ya sea a foros o encuestas.

3.3 Diagrama de clases.

Un diagrama de clases es una colección estática de los elementos del modelo, como clases, tipos y sus relaciones, conectados unos a otros y a sus contenidos.

Teniendo en cuenta que el sistema es un módulo genérico que sirve para cualquier tipo de proyecto y el CMS para el cual fue creado, cuenta como mínimo con 57 tablas de la base datos, algunos de sus elementos no fueron modelados con los elementos estándares de UML, solo fueron representadas las que se utilizaron como medio para la recuperación de datos estadísticos. Todas estas clases que se muestran a continuación son persistentes, o sea, corresponden a entidades proveedoras de información.

Las clases persistentes son aquellas que necesitan almacenar su estado en un medio permanente. Estas clases modelan la información y el comportamiento asociado de algún fenómeno o concepto, como una persona, un objeto o suceso del mundo real. El diagrama de clases persistente ayuda a comprender de qué información depende el sistema.

En el Anexo 5 se puede ver el Diagrama de Clases:

3.4 Modelo de datos

El modelo de datos describe la representación lógica y física de los datos persistentes en el sistema. Se utiliza para definir la correspondencia entre las clases de diseño persistentes y la estructura de datos persistente. Es frecuentemente necesitado cuando se tiene un modelo de objetos y el mecanismo de almacenamiento se basa en una base de datos relacional.

En este caso, el modelo de datos se ha obtenido partiendo del diagrama de clases persistentes. Fue necesario para esto introducir algunos cambios debido a consideraciones que se tuvieron en cuenta.

Teniendo en cuenta los principios generales del diseño de bases de datos se realizaron alteraciones en algunas tablas de la base datos manteniendo la integridad y la normalización de la misma.

El cambio principal fue que se le añadieron dos campos dinámicos a la tabla accesslog (navegador, sistema operativo) con el objetivo de almacenar el navegador que utilizó el usuario para visitar el sitio y el sistema operativo que usa en ese momento. Esto viene dado por la necesidad de almacenar esta información.

3.5 Diagrama Entidad Relación de la BD.

Un diagrama entidad-relación muestra las distintas relaciones entre las entidades con el objetivo de poder filtrar datos de un relación que estén vinculadas con otras relaciones. Existen varios tipos de relaciones como por ejemplo relaciones de generalización/especialización o herencia. La herencia es un intento de adaptación de estos diagramas al paradigma orientado a objetos. La herencia es un tipo de relación entre una entidad "padre" y una entidad "hijo". La entidad "hijo" hereda todos los atributos y relaciones de la entidad "padre" , uno a uno la cual se basa en que la llave primaria de una entidad, cualquiera que sea, pasa a ser foránea en la otra relación , la de uno a muchos donde la llave primaria de la primera relación pasa a ser referencia de la otra y por último la de muchos a muchos donde se forma una relación nueva con las llaves de cada una de las entidades que se relacionan y los atributos propios de la relación. Todo esto se hace con el objetivo de que la base datos esté en un estado consistente a la hora de la recuperación e inserción de datos.

Formalmente, los diagramas Entidad-Relación son un lenguaje gráfico para describir conceptos. Informalmente, son simples dibujos o gráficos que describen la información que trata un sistema de información y el software que lo automatiza.

En el Anexo 6 se muestra el diagrama Entidad-Relación.

Entre el diseño lógico y físico de la base de datos no hay grandes diferencias, sólo se hicieron algunas consideraciones para garantizar algunos elementos de diseño y facilitar la programación en algunos casos. Los cambios que se proponen no afectan la integridad de los datos, solo provocan un incremento en el tamaño de las tablas que se convierte en un aumento de velocidad ya que se reducen el número de consultas a la base de datos. Ante este compromiso entre capacidad de memoria y velocidad se decidió sacrificar lo primero considerando que para el lugar en el que se explotará el sistema, esto no es una limitante.

3.5.1 Descripción de las tablas.

Nombre: accesslog		
Descripción: En esta tabla se almacenan los datos referentes a las páginas, IP de acceso, tiempo de acceso, dirección .etc.		
Atributo	Tipo	Descripción
aid	int	Guarda el número de acceso que le corresponde a esa página el cual permite identificarla.
sid	varchar	
title	varchar	Almacena el título del ítem a través del cual se accede a la URL de la página.
path	varchar	Almacena la URL de la página a la que se accede.
url	varchar	Almacena la URL que se muestra en el navegador.
hostname	varchar	Almacena el IP de donde se accedió al sitio.
uid	int	Almacena el identificador de usuario que accedió al sitio.
timer	int	Almacena el tiempo de acceso a la página.
timestamp	int	Almacena la fecha y hora de acceso a la página.
navegador	varchar	Almacena el navegador con el que se accede al sitio o a la página.
sistema_operativo	varchar	Almacena el sistema operativo que se esta usando para navegar por el sitio

Nombre: users		
Descripción: En esta tabla se almacenan los datos referentes a los usuarios		
Atributo	Tipo	Descripción

uid	int	Identifica al usuario registrado.
name	varchar	Nombre del usuario.
pass	varchar	Almacena el password del usuario.
mail	varchar	Almacena el correo electrónico del usuario.
created	int	Almacena la fecha de creación del usuario.
access	int	Almacena la fecha de acceso del usuario al sitio.
login	int	Almacena la hora en que se loguea el usuario.
status	tinyint	Guarda 0 si el usuario se desconecta y 1 si está conectado.
timezone	int	Almacena la zona horaria.
language	varchar	Lenguaje que se está utilizando en el sitio.
init	varchar	El correo del usuario.
data	longtext	Almacena las operaciones realizada por el usuario.

Nombre: node		
Descripción: En esta tabla se almacenan todo lo referente a nodos creados.		
Atributo	Tipo	Descripción
nid	int	Identifica al nodo creado.
vid	int	Nombre del usuario que creó el nodo.
type	varchar	Almacena el tipo de nodo creado.
title	varchar	Almacena el título del nodo creado por el usuario.
uid	int	Identificador del usuario que creó el nodo.
created	int	Almacena la fecha que se creó el nodo.
login	int	Almacena la hora en que se loguea el usuario.
status	tinyint	Guarda 0 si el usuario se desconecta y 1 si está conectado.
timezone	int	Almacena la zona horaria.
language	varchar	Lenguaje que se está utilizando en el sitio.
init	varchar	El correo del usuario.
data	longtext	Almacena las operaciones realizada por el usuario.

Nombre: comment		
Descripción: En esta tabla se almacenan todo lo referente a comentarios hechos en el sitio.		
Atributo	Tipo	Descripción
cid	int	Identifica al comentario.
pid	int	

nid	int	Almacena el nodo al cual fue hecho el comentario.
uid	int	Almacena el usuario que hizo el comentario.
subject	varchar	Asunto del comentario.
comment	varchar	Texto del comentario.
hostname	varchar	Equipo de donde se hizo el comentario.
timestamp	int	Fecha en que fue hecho el comentario.
name	varchar	Nombre del usuario que hizo el comentario.

Nombre: system		
Descripción: En esta tabla se almacenan los datos de los módulos y temas		
Atributo	Tipo	Descripción
filename	varchar	Almacena la dirección y nombre del módulo.
name	varchar	Almacena el nombre del módulo.
type	varchar	Almacena el tipo de archivo.
description	varchar	Almacena la descripción del módulo.
status	int	Almacena el estado en que se encuentra el módulo (1 si está activado y 0 si está desactivado).

Capítulo 4: Implementación y Prueba

En el presente capítulo se desarrollan los flujos de trabajo de implementación y prueba. Se muestra la situación física de los componentes lógicos desarrollados a través del modelo de despliegue. El modelo de despliegue es un modelo de objetos que describe la distribución física del sistema entre los nodos de cómputo.

Cada nodo representa un recurso de cómputo, normalmente un procesador o un dispositivo de hardware similar. Se sitúa el software en el hardware que lo contiene. En este caso la aplicación se encuentra hospedada en un servidor Web, la misma se comunica con un sistema de gestión de base de datos (MySQL) que se encuentra en otro servidor.

En el despliegue del sistema se utilizó una máquina cliente que se comunicará con el servidor web vía http donde estará el módulo estadístico y este se conecta a una base de datos que estará alojada en otro servidor por motivos de seguridad.

En el Anexo 7 se muestra el diagrama de despliegue correspondiente al sistema.

4.1 Diagrama de componentes.

El diagrama de componentes define cómo las clases, artefactos y otros elementos de bajo nivel, se unen para formar componentes de alto nivel y las conexiones entre ellos. Son artefactos de software compilados que trabajan acoplados para brindar el comportamiento requerido dentro de las restricciones definidas en el proceso de captura de requisitos.

En el Anexo 8 se muestra el diagrama de componentes correspondiente al sistema.

4.2 Prueba

El desarrollo del software implica una serie de actividades de producción en las que las posibilidades de que aparezca la falibilidad humana son comunes. Los errores pueden empezar a darse desde el primer

momento del proceso en el que los objetivos pueden estar especificados de forma errónea e imperfecta; así en los posteriores pasos del diseño y desarrollo. Debido a la imposibilidad humana de trabajar y comunicarse de forma perfecta, el desarrollo del software ha de ir acompañado de una actividad que garantice la calidad.

Las pruebas son una actividad en la cual un sistema o componente es ejecutado bajo ciertas condiciones o requerimientos especificados, los resultados son observados y registrados, y una evaluación es hecha de algún aspecto del sistema o componente.

La prueba de software es un elemento crítico para la garantía de la calidad del software y representa una revisión final de las especificaciones del diseño y de la codificación.

En el caso del módulo se llevaron a cabo solamente pruebas de funcionamiento del software

4.2.1 Normas de prueba

1. La prueba es un proceso de ejecución de un programa con la intención de descubrir un error.
2. Un buen caso de prueba es aquel que tiene una alta probabilidad de mostrar un error no descubierto hasta entonces.
3. Una prueba tiene éxito si descubre un error no detectado hasta entonces.

Caso de uso: Escoger Opciones.

Entrada	Resultados	Condiciones
El administrador debe pulsar sobre el botón "habilitado o desactivado" para habilitar o deshabilitar el proceso de control estadístico al sitio.	Si el administrador no habilita la opción "habilitado" Este no podrá visualizar ningún dato estadístico.	El administrador debe tener la completa seguridad que habilitó esa opción.

Prueba	Descripción
nodo	En la base datos debe haber sido guardado 0 si no fue publicado y 1 si fue publicado.
navegador	Registre en la base datos el navegador que se utiliza.
IP	Almacene el IP de forma correcta.
datos del sistema(página mas visitada)	Si todavía el sitio no ha recibido ninguna visita diferente del administrador saldrá un mensaje de "No se han efectuado visitas al sitio".
representación gráfica	Se realizaron comprobaciones de correspondencia de los datos con respecto a la gráfica.
usuario	En caso de ser usuario anónimo guarde el IP.

Se llevaron a cabo pruebas de seguridad con el objetivo de asegurarse que el sistema solo era accedido por los actores deseados, en este caso por el administrador, el único que contaba con todos los permisos necesarios para visualizar los registros estadísticos, así como pruebas de volumen enfocadas a verificar las habilidades de la herramienta para manejar grandes cantidades de datos residentes en la base datos.

De modo general se hicieron algunas pruebas, se instaló el módulo en distintas máquinas del proyecto para probar el funcionamiento del mismo y así llegar a la conclusión de si se había llegado o no a los objetivos trazados. Se comprobó si se estaban registrando bien todos los datos respecto a los usuarios y los accesos a páginas del sitio unido a su visualización gráfica para así tener una idea del posible mejoramiento de la aplicación.

En el sistema de validación para el tratamiento de errores fueron referidos principalmente a errores de interacción del usuario del sistema que son aquellos en los que se puede incurrir a falta de conocimiento o experiencia de la explotación del sistema.

En este capítulo se han descrito actividades y artefactos correspondientes a los flujos de trabajo de Diseño y de Implementación de RUP, obteniéndose como resultados de los mismos el Modelo de Clases, el Modelo de Despliegue y el Modelo de Implementación. Estos resultados son importantes para la implementación del sistema estadístico ya que rigen todos los procesos de desarrollo de software a la hora de implementar el módulo. Se hicieron cada una de las relaciones entre clases que interactúan con

el sistema así como el diagrama de entidad-relación para mostrar las dependencias y referencias entre las tablas de la base datos que se utilizaron a la hora de desarrollar el sistema. En el despliegue se llevaron a cabo los distintos tipos de comunicaciones que existen entre los dispositivos físicos de la aplicación y en el de componentes se controlan cada uno de los ficheros, ejecutables, páginas y archivos que fueron utilizados, así como la relación entre cada uno de estos para la ejecución del módulo.

Conclusiones generales.

Con el desarrollo de este trabajo, se considera que han sido cumplidos los objetivos propuestos, ya que:

Se elaboró un modelo, donde se definieron cuatro flujos de trabajo fundamentales, que permiten integrar la gestión estadística de una web al proceso de desarrollo de software desde el inicio hasta las pruebas del sistema, de una forma continua y controlada.

Para cada uno de los flujos se detallaron los artefactos generados y establecidos por RUP.

El modelo es fácilmente comprensible para cualquier equipo de desarrolladores que lleven a cabo el control estadístico de su web, hija de este CMS, tomándolo como guía durante el proceso de gestión. Para lograr este objetivo se representaron los artefactos generados mediante diagramas y tablas que de cierta forma hacen el análisis de los mismos mucho más sencillo.

Además de los elementos mencionados anteriormente, durante el desarrollo de este trabajo se arribaron a un conjunto de conclusiones que poseen una marcada importancia, las mismas se relacionan a continuación:

1. A pesar de que este modelo constituye una guía para gestionar de manera controlada e iterativa el control estadístico de la web montada sobre Drupal no representa por ello un marco rígido a implantar, sino que por el contrario, puede ser adaptado a las características y necesidades específicas de cada proyecto en particular.
2. Aunque quizás este módulo de gestión estadística no esté optimizado en su totalidad, ha demostrado ser una opción factible, por lo que está despertando un gran interés entre los integrantes del proyecto.
3. Finalmente, se considera que la utilización del modelo propuesto contribuirá a desarrollar soluciones de software en la UCI que cumplan con los requisitos exigidos por los clientes, al permitir llevar un control continuo y efectivo de las estadísticas.

Recomendaciones.

En el desarrollo de un trabajo siempre quedan un conjunto de aspectos relevantes, que por cuestiones muy diversas, no siempre pueden ser analizados o al menos con la profundidad que se requiere. Este trabajo no queda exento de ello; por lo que a continuación se relacionan un conjunto de ideas que se consideran necesarias para darle continuidad al mismo, contribuyendo a un modelo más acabado:

- Implementar un mejor estilo a las páginas que muestran el contenido estadístico de acuerdo con las necesidades específicas de su proyecto.
- Añadir la funcionalidad en cuanto a los IP de donde visitan el sitio a nivel internacional.
- Implementar el módulo para versiones superiores a la versión 4.7 de Drupal.
- A todos los que de una forma u otra utilizan Drupal como CMS para el desarrollo de aplicaciones web y deseen llevar el control estadístico del mismo.
- Que el sistema desarrollado se retroalimente, constantemente, de los resultados obtenidos en su aplicación en proyectos de desarrollo de software, de forma tal que muchos de estos resultados constituyan una fuente importante de información a reutilizar en procesos posteriores con características similares.
- Añadir al módulo un contador de líneas de código (LOC) de forma que se puedan incorporar las métricas asociadas al tamaño del proyecto en LOC.
- Hacerlo más personalizable de modo que el usuario escoja la gráfica que desee visualizar para la salida de sus datos.

Referencias Bibliográficas.

1. GISBERT, J. *La importancia de las estadísticas*. 2007, Disponible en: <http://MundodelMarketing.com>.
2. YANOVER, A. J. API - ¿Qué significa API? 2007, nº Disponible en: <http://www.seohome.com/glosario/api.html>.
3. ANÓNIMO. 2007, nº Disponible en: <http://es.wikipedia.org/wiki/PHP>.
4. ---. MySql. 2007, nº Disponible en: <http://es.wikipedia.org/wiki/MySQL>.

Bibliografía

JACOBSON, Ivar; RUMBAUGH, James, BOOCH, Grady, "El lenguaje unificado de modelado".
<http://bibliodoc.uci.cu/pdf/reg03050.pdf>, 2000.

PRESSMAN., R. S. Tomado de la biblioteca UCI. Ingeniería del software. Un enfoque práctico.
<http://bibliodoc.uci.cu/pdf/reg02689.pdf>.

SCHMULLER, J. Tomado de la biblioteca UCI. Aprendiendo UML en 24 horas.
<http://bibliodoc.uci.cu/pdf/reg00004.pdf>, 2000.

Jacobson, Ivar; Booch, Grady; Rumbaugh, James, "Proceso unificado del Software". education.S.A,
Madrid, 2000

ANÓNIMO. 2007, Disponible en: <http://api.drupal.org/>.

ANÓNIMO. Publicado el: 24/4/2005, última actualización: 24/4/2007. Disponible en:
http://www.washeebo.com/sargento/03_php/0318/0318.php.

González y Kurszewski, José, última actualización: 19/5/2007. Disponible en:
http://www.washeebo.com/sargento/03_php/03_php.php.

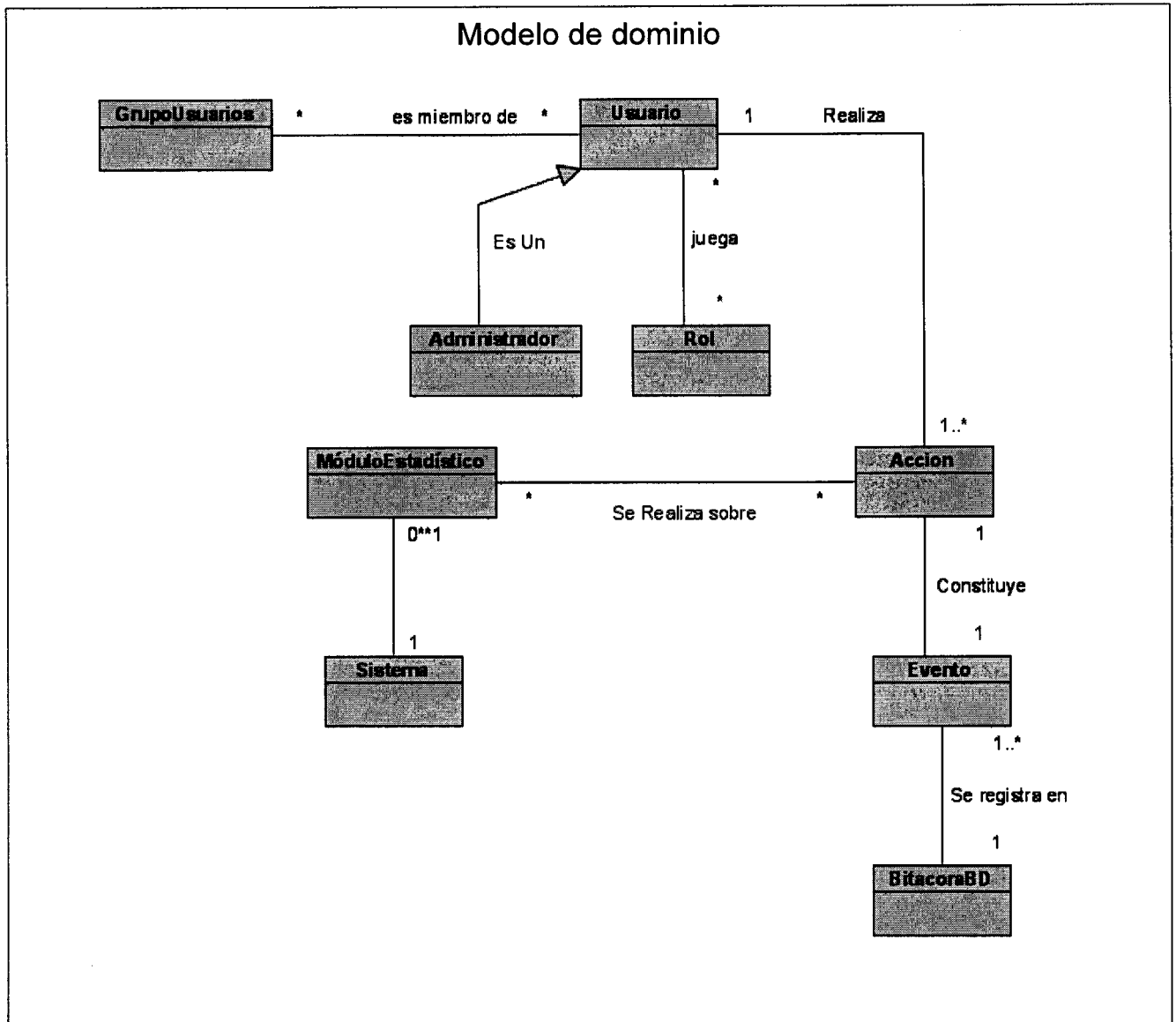
ANÓNIMO. 2007, nº Disponible en: <http://www.php.net/manual/en/function.gd-info.php>.

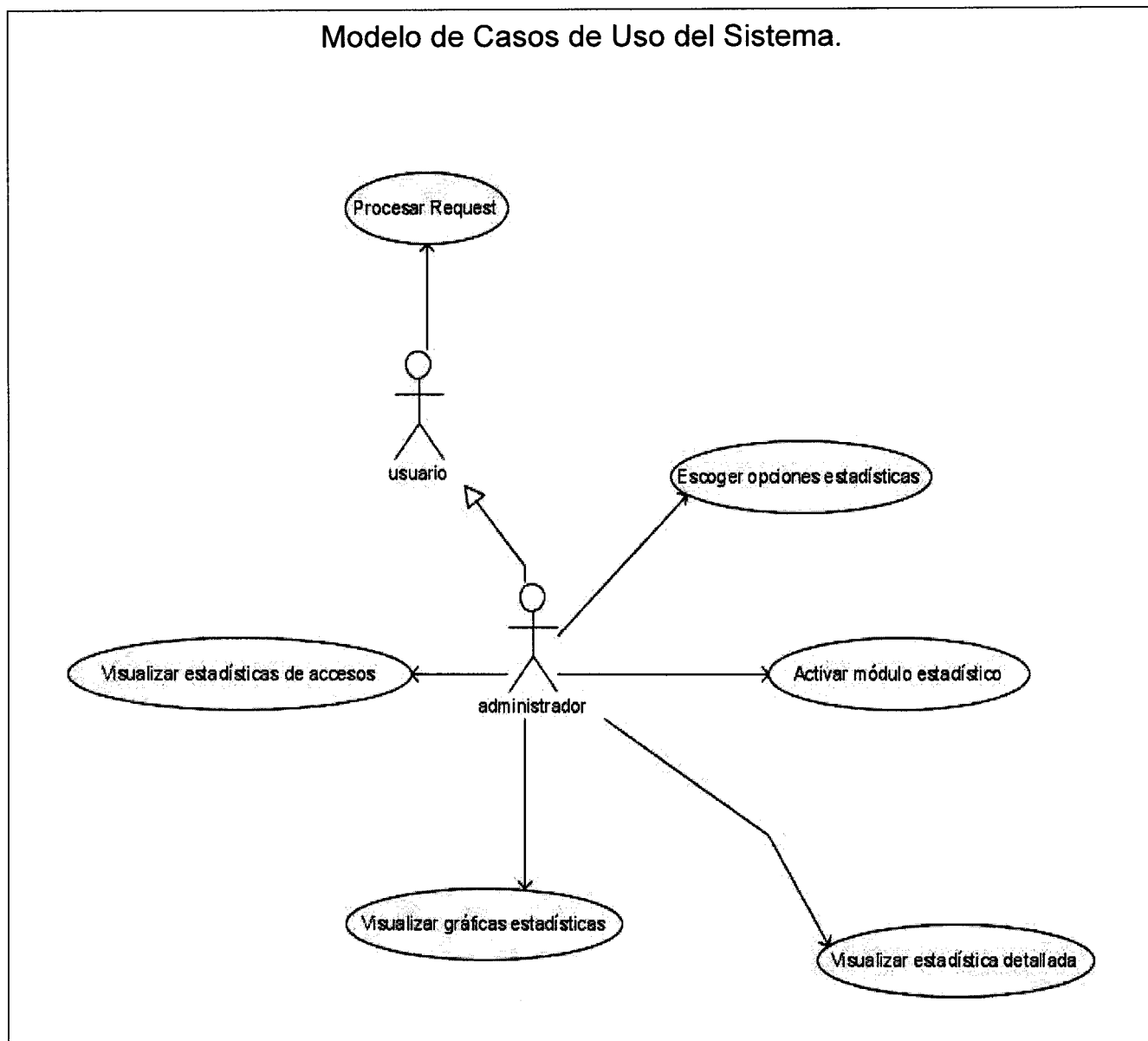
VÁZQUEZ, J. A. G. Tomado biblioteca UCI. Desarrollo Web con PHP y MySQL.
<http://bibliodoc.uci.cu/pdf/reg02138.pdf>, 2003.

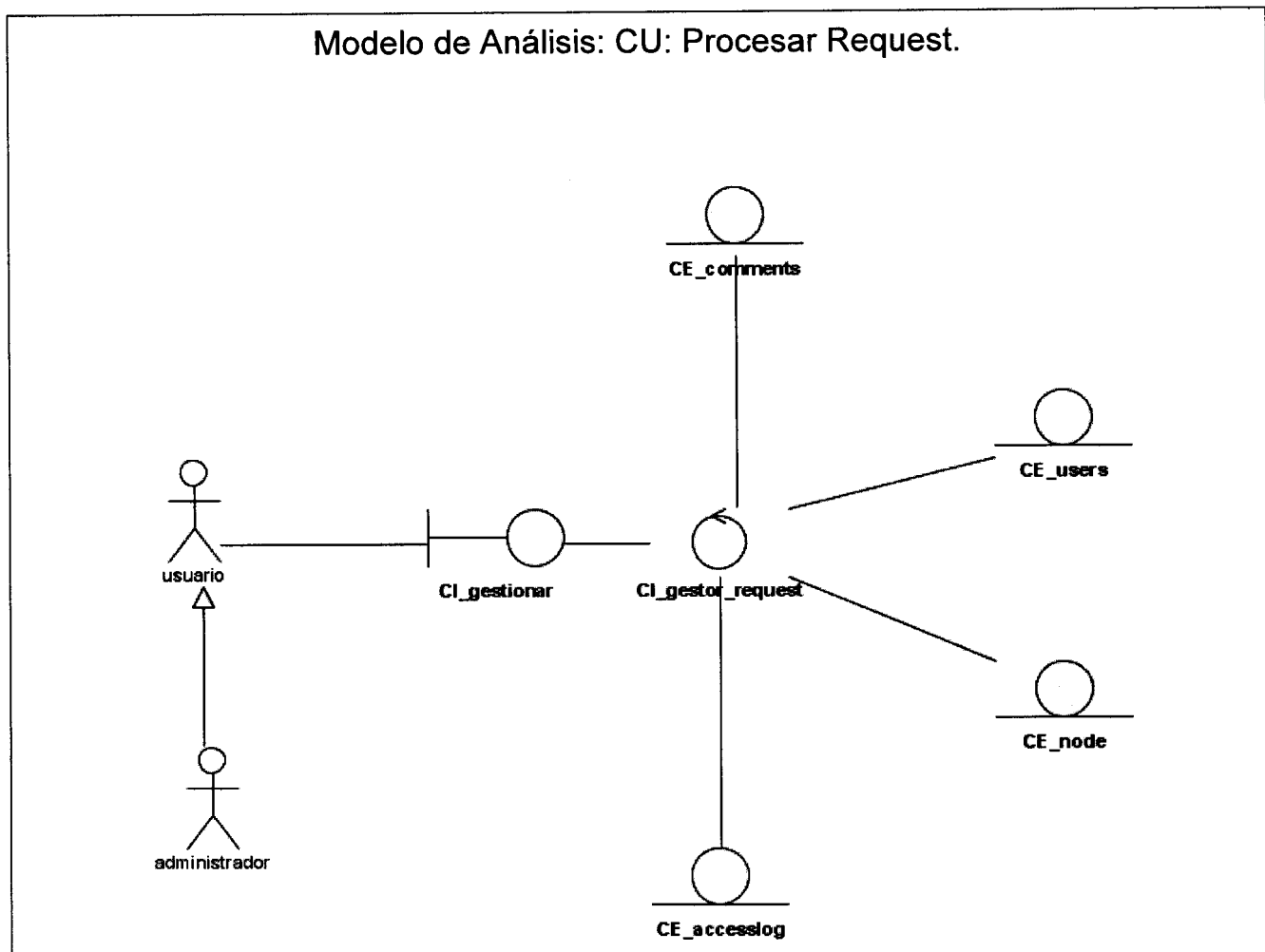
Portal oficial de Drupal. <http://drupal.org.es/>

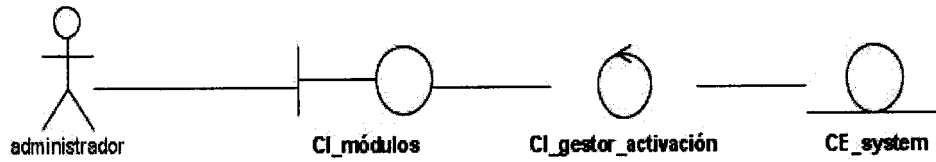
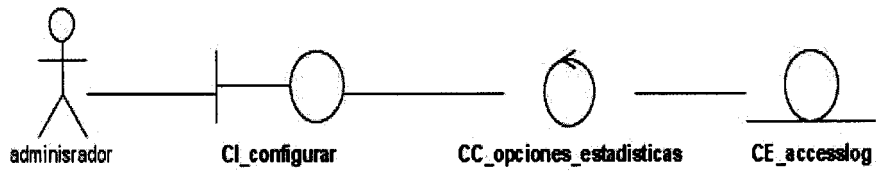
Anexos

Anexo 1: Modelo del dominio.

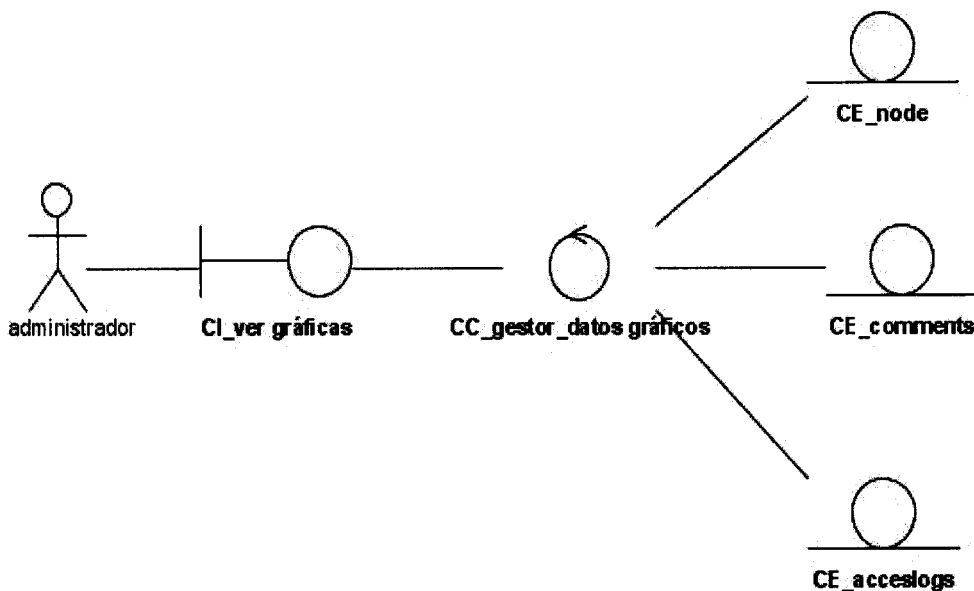


Anexo 2: Modelo de Casos de uso del sistema.

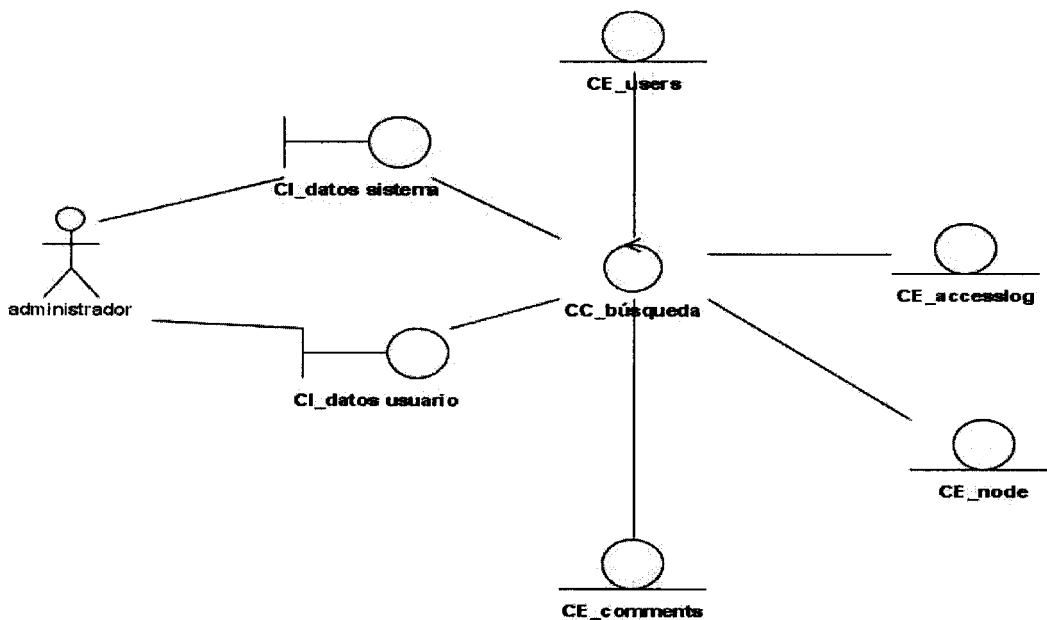
Anexo 3: Modelo de análisis.

Modelo de Análisis: CU: Activar módulo estadístico**Modelo de Análisis: CU: Escoger opciones estadísticas.**

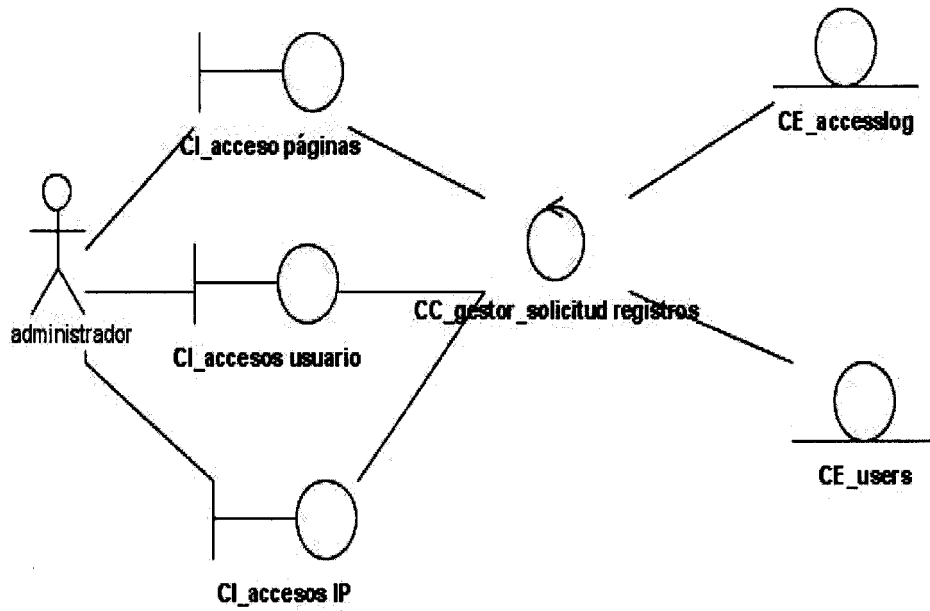
Modelo de Análisis: CU: Visualizar gráficas estadísticas.



Modelo de Análisis: CU: Visualizar estadística detallada.



Modelo de Análisis: CU: Visualizar estadísticas de accesos.



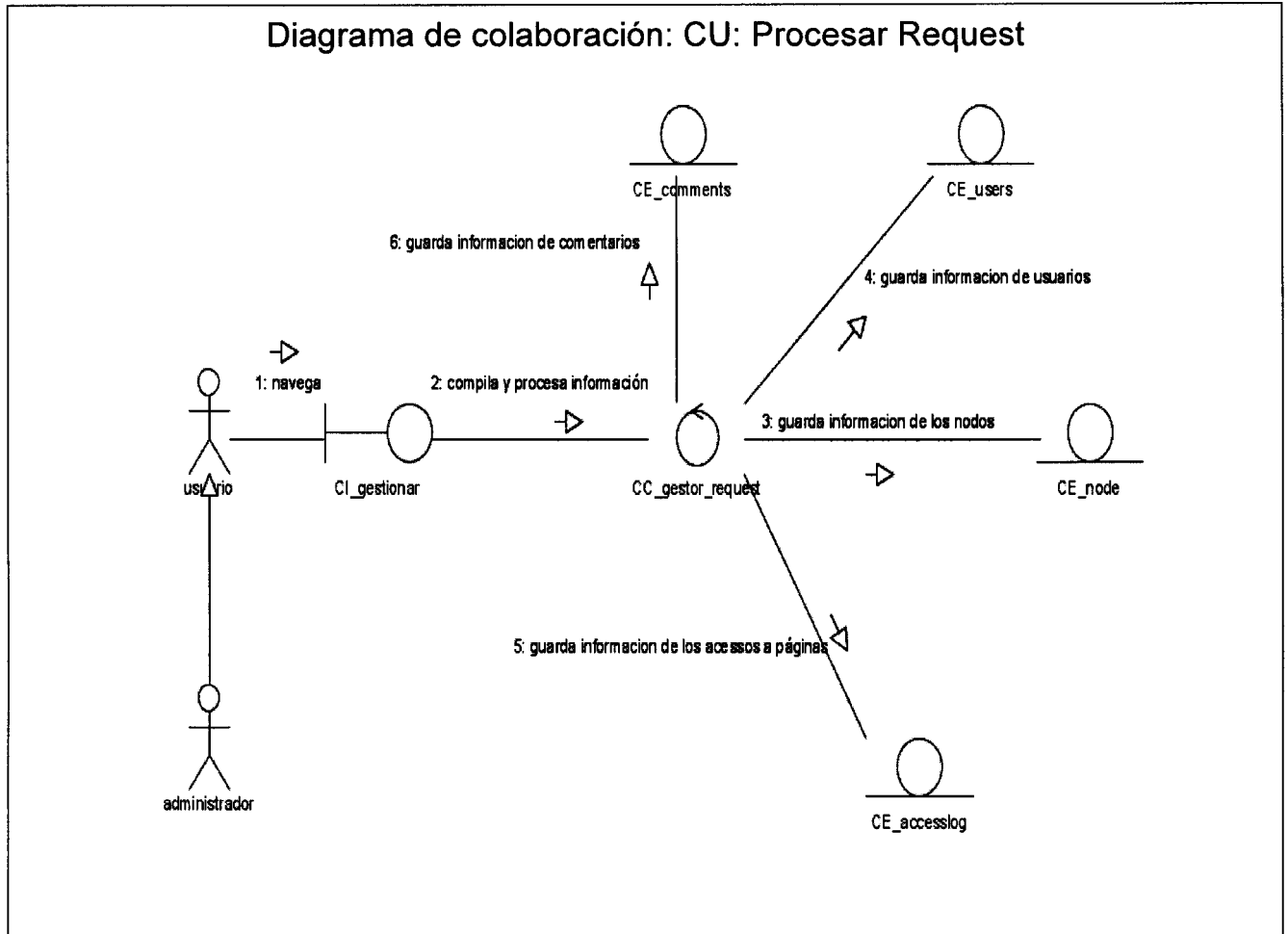
Anexo 4: Diagramas de colaboración.

Diagrama de colaboración: CU: Activar Módulo estadístico

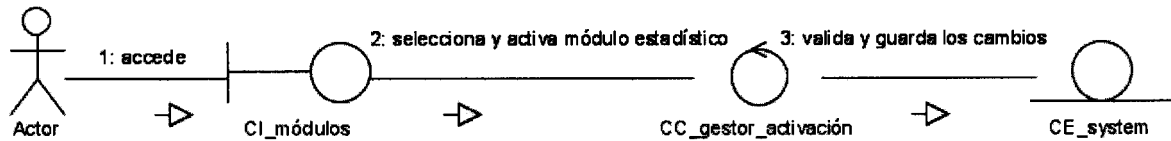


Diagrama de colaboración: CU: Escoger opciones estadísticas.

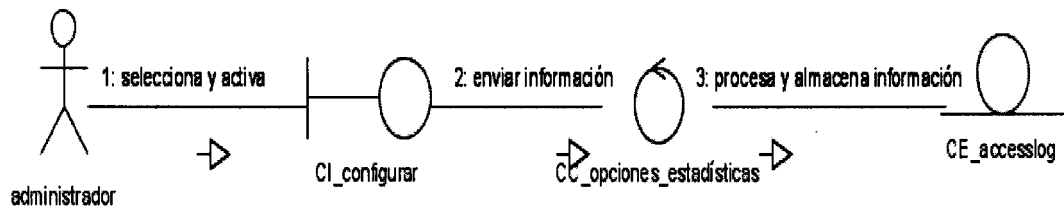


Diagrama de colaboración: CU: Visualizar gráficas estadísticas.

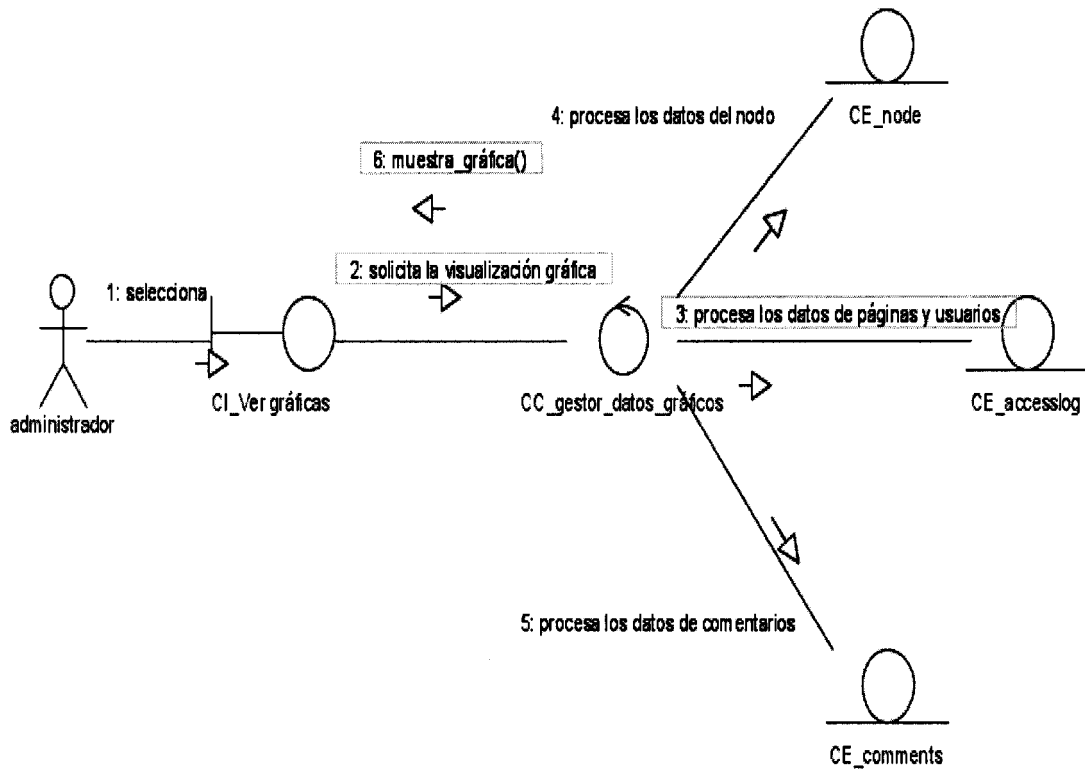


Diagrama de colaboración: CU: Visualizar estadística detallada.

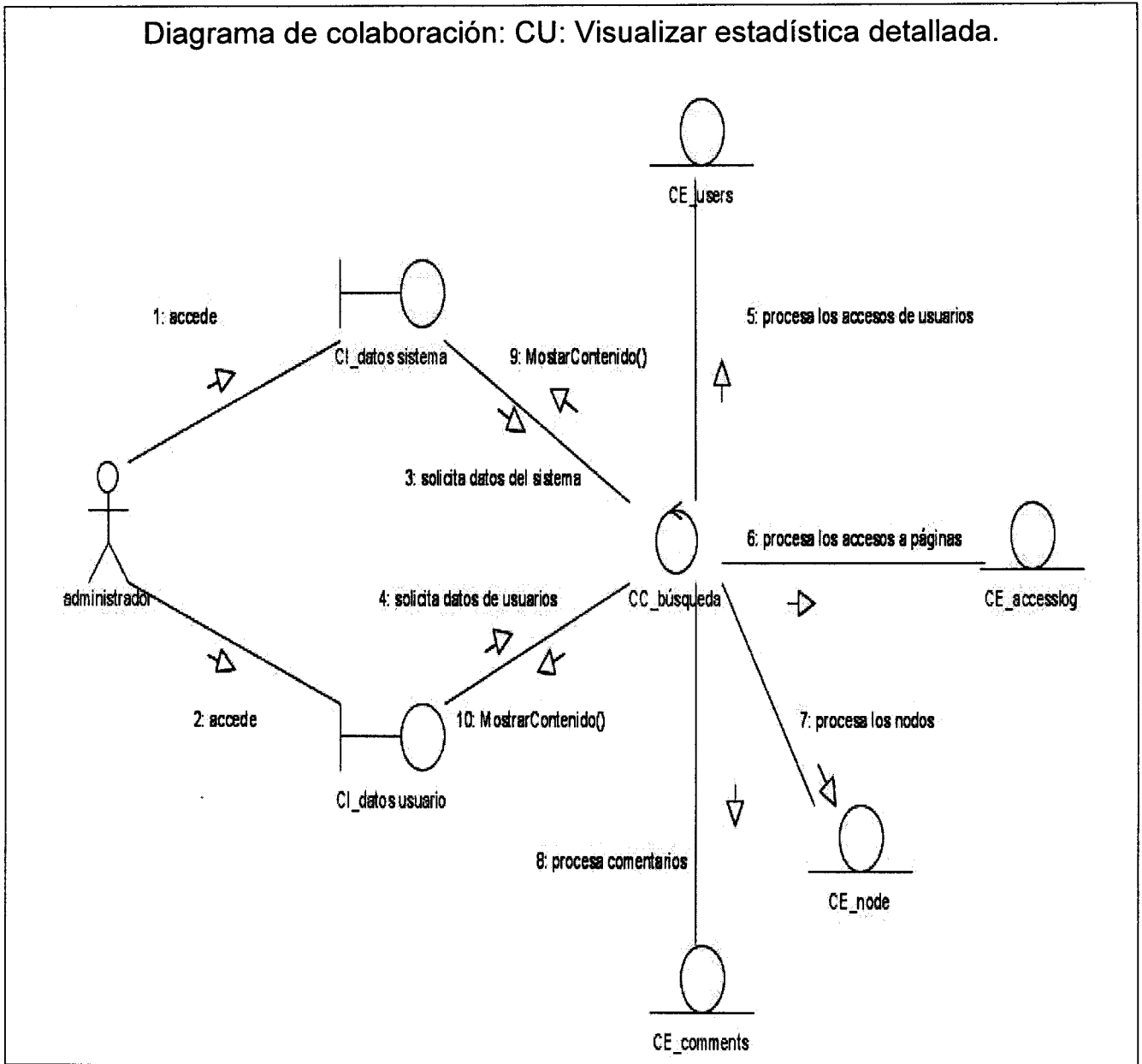
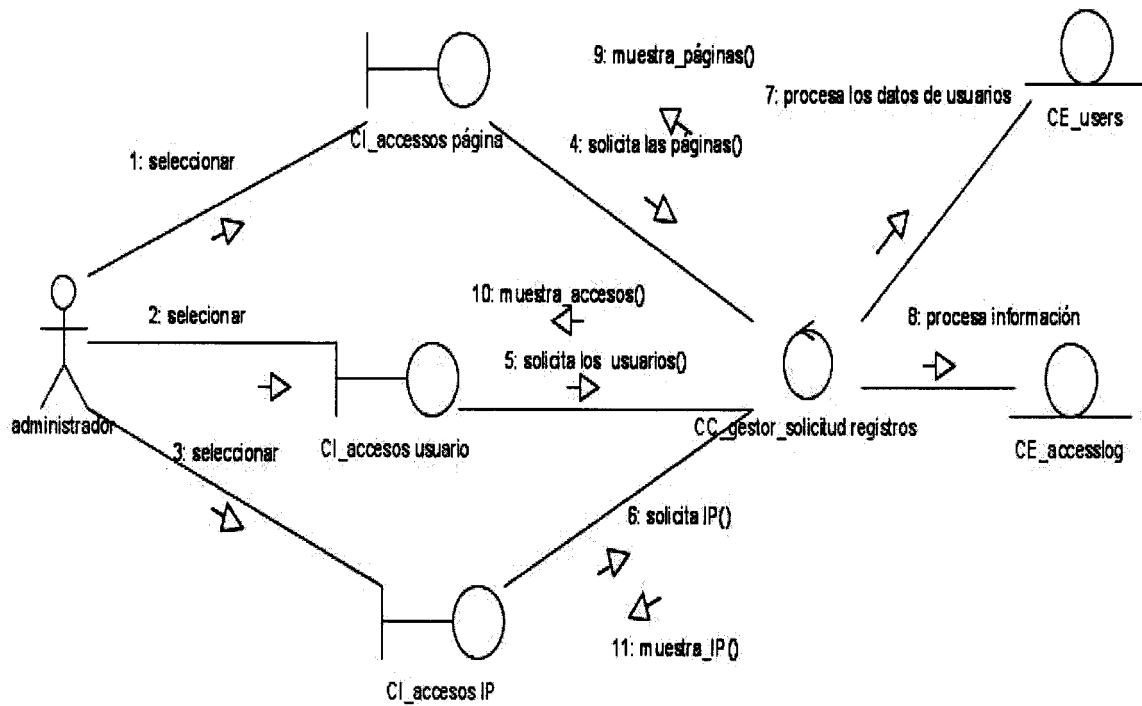
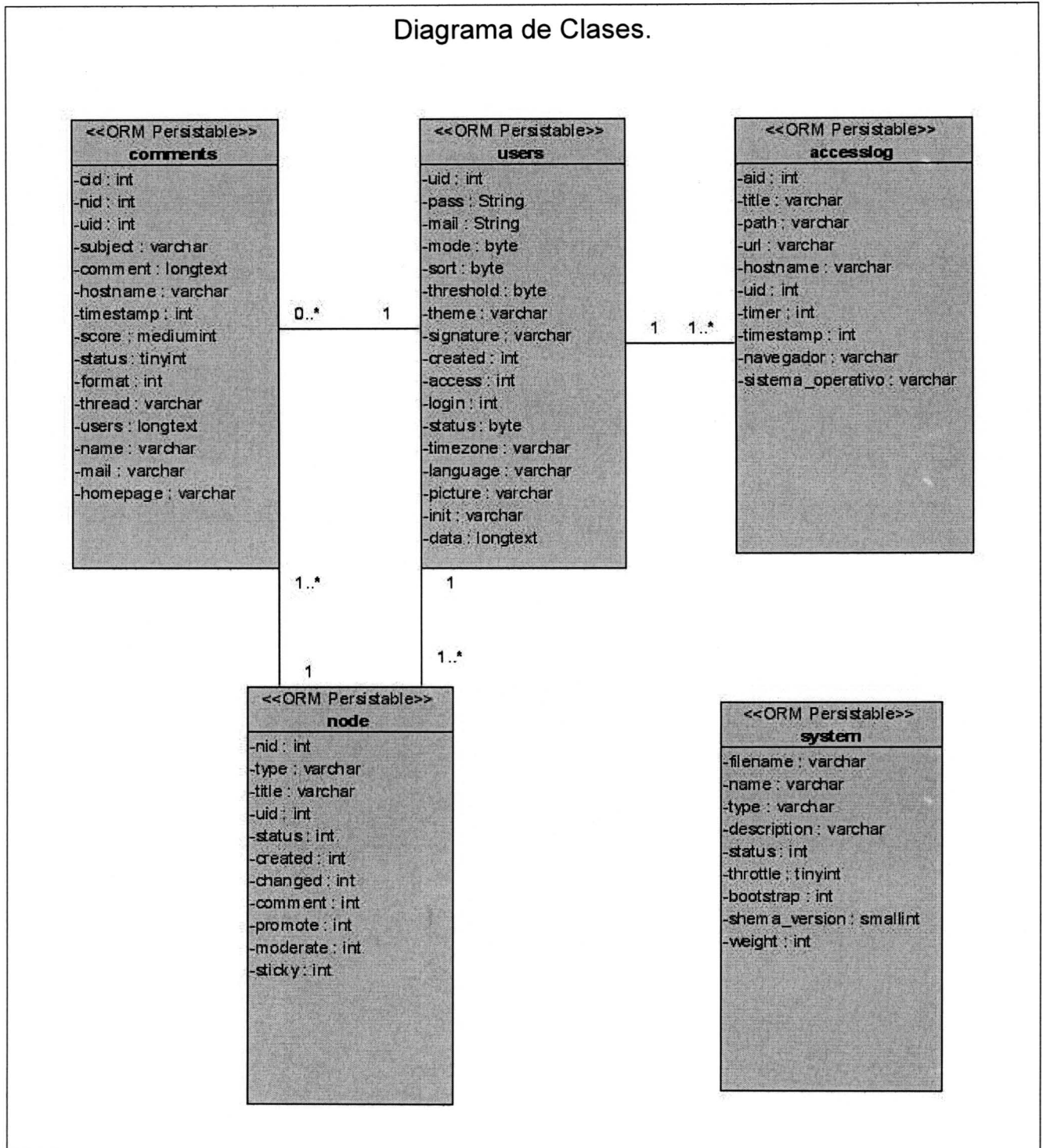


Diagrama de colaboración: CU: Visualizar estadísticas de accesos.



Anexo 5: Diagrama de clases.



Anexo 6: Diagrama Entidad-Relación.

Diagrama Entidad-Relación.

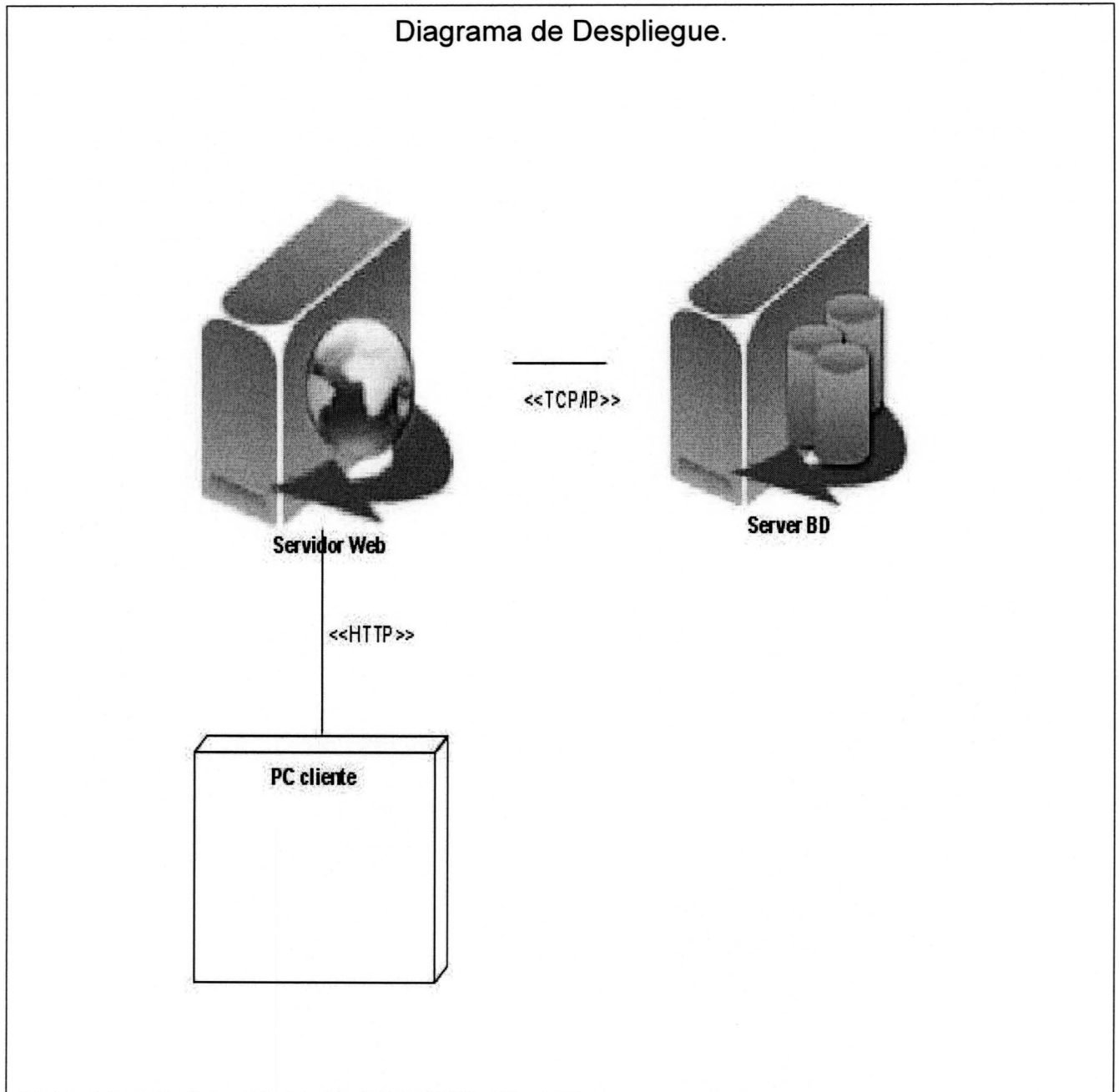
comments		
+cid	integer(10)	Nullable = false
#nid	integer(10)	Nullable = false
#uid	integer(10)	Nullable = false
subject	varchar(64)	Nullable = false
comment	char(255)	Nullable = false
hostname	varchar(255)	Nullable = false
timestamp	integer(11)	Nullable = false
score	smallint(5)	Nullable = false
status	tinyint(3)	Nullable = false
format	integer(4)	Nullable = false
thread	varchar(255)	Nullable = false
users	char(255)	Nullable = true
name	varchar(60)	Nullable = true
mail	varchar(64)	Nullable = true
homepage	varchar(255)	Nullable = true

users		
+uid	integer(10)	Nullable = false
pass	varchar(255)	Nullable = true
mail	varchar(255)	Nullable = true
mode	tinyint(3)	Nullable = false
sort	tinyint(3)	Nullable = false
threshold	tinyint(3)	Nullable = false
theme	varchar(255)	Nullable = false
signature	varchar(255)	Nullable = false
created	integer(11)	Nullable = false
access	integer(11)	Nullable = false
login	integer(11)	Nullable = false
status	tinyint(4)	Nullable = false
timezone	varchar(8)	Nullable = true
language	varchar(12)	Nullable = false
picture	varchar(255)	Nullable = false
init	varchar(64)	Nullable = true
data	char(255)	Nullable = true

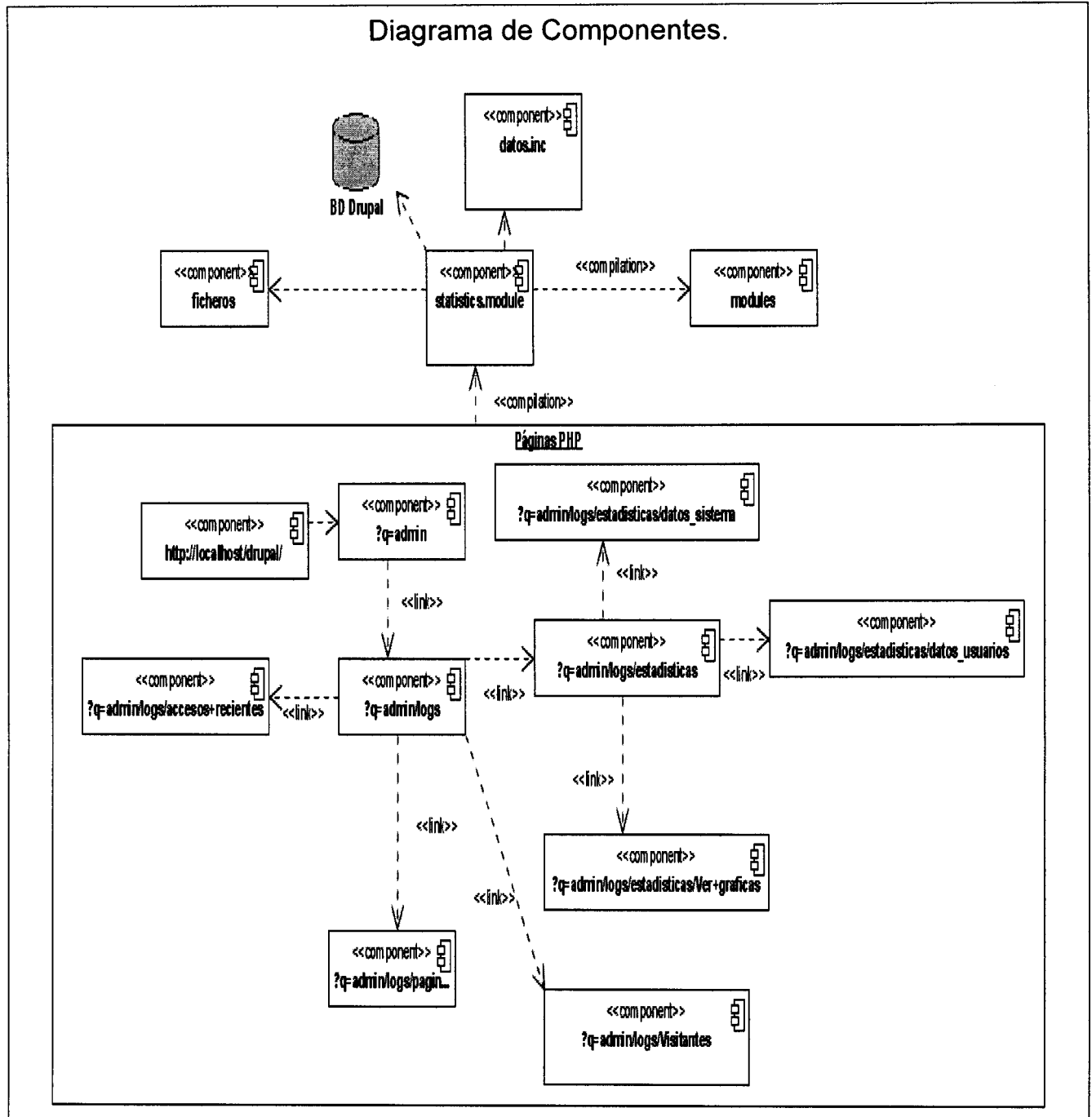
node		
+nid	integer(10)	Nullable = false
vid	integer(10)	Nullable = false
type	varchar(32)	Nullable = false
title	varchar(128)	Nullable = false
#uid	integer(10)	Nullable = false
status	integer(4)	Nullable = false
created	integer(11)	Nullable = false
changed	integer(11)	Nullable = false
comment	integer(2)	Nullable = false
promote	integer(2)	Nullable = false
moderate	integer(2)	Nullable = false
sticky	integer(2)	Nullable = false

accesslog		
+aid	integer(10)	Nullable = false
sid	varchar(32)	Nullable = false
title	varchar(255)	Nullable = true
path	varchar(255)	Nullable = true
url	varchar(255)	Nullable = true
hostname	varchar(128)	Nullable = true
#uid	integer(10)	Nullable = false
timer	integer(10)	Nullable = false
timestamp	integer(10)	Nullable = false
navegador	varchar(255)	Nullable = true
sistema_operativo	varchar(255)	Nullable = true

system		
+filename	varchar(255)	Nullable = false
name	varchar(255)	Nullable = false
type	varchar(255)	Nullable = false
description	varchar(255)	Nullable = false
status	integer(2)	Nullable = false
throttle	tinyint(1)	Nullable = false
bootstrap	integer(2)	Nullable = false
shema_version	smallint(3)	Nullable = false
weight	integer(2)	Nullable = false

Anexo 7: Diagrama de despliegue.

Anexo 8: Diagrama de Componentes.



Glosario de términos

Abstracción: Una abstracción se refiere a cómo se representan determinados problemas en el espacio del programa. Los lenguajes de programación en si mismos proporcionan abstracciones.

Bugs: Un error de software o computer bug, que significa bicho de computadora, es el resultado de una falla de programación introducida en el proceso de creación de programas de computadora.

Caché: es un conjunto de datos duplicados de otros originales, con la propiedad de que los datos originales son costosos de acceder, normalmente en tiempo, respecto a la copia en el caché. Cuando se accede por primera vez a un dato, se hace una copia en el caché; los accesos siguientes se realizan a dicha copia, haciendo que el tiempo de acceso medio al dato sea menor.

CGI: Es una importante tecnología de la World Wide Web que permite a un cliente (explorador web) solicitar datos de un programa ejecutado en un servidor web. CGI especifica un estándar para transferir datos entre el cliente y el programa.

Dominio: Es un nombre de equipo que proporciona nombres más fácilmente recordados en lugar de la IP numérica. Permiten a cualquier servicio moverse a otro lugar diferente en la topología de Internet, que tendrá una dirección IP diferente. Conjunto de máquinas que responden a una misma subred.

E-administración: La prestación de todos los servicios públicos, haciendo uso de la interactividad, rapidez, eficiencia y personalización que ofrecen las nuevas tecnologías.

Encapsulamiento: Es la llamada ocultación de la información, es la capacidad de ocultar los procesos internos de un objeto a sus usuarios y proporcionar una interfaz sólo para los miembros que quiera que el cliente tenga posibilidad de manipular directamente.

Hardware: Conjunto de elementos materiales que componen un ordenador. En dicho conjunto se incluyen los dispositivos electrónicos y electromecánicos, circuitos, cables, tarjetas, armarios o cajas, periféricos de todo tipo y otros elementos físicos.

Llave primaria: Es aquel atributo el cual hace que cada registro de la entidad debe estar separado y ser unívocamente identificable del resto de los registros de esa misma entidad; y quien permite esta identificación.

LDAP: Es un servicio de directorio ordenado y distribuido para buscar diversa información en un entorno de red. LDAP puede considerarse una base de datos.

Multiplataforma: Es un término usado para referirse a los programas, sistemas operativos, lenguajes de programación, u otra clase de software, que puedan funcionar en diversas plataformas.

Nodo: Se define como nodo a cada contenido o página que se crea en Drupal, también es cualquier dispositivo físico en el diagrama de despliegue. En general es un objeto que se puede relacionar con otros objetos.

Paradigma: Un paradigma está constituido por los supuestos teóricos generales, las leyes y las técnicas para su aplicación que adoptan los miembros de una determinada comunidad científica.

Patrones: Son la base para la búsqueda de soluciones a problemas comunes en el desarrollo de software y otros ámbitos.

Polimorfismo: Funcionalidad que permite al código antiguo invocar nuevo código.

Rol: Papel que ejerce un actor en una actividad o proyecto.

Soap: Es un protocolo estándar creado por Microsoft, IBM y otros, está actualmente bajo el auspicio de la W3C que define cómo dos objetos en diferentes procesos pueden comunicarse por medio de intercambio de datos XML. SOAP es uno de los protocolos utilizados en los servicios Web.

Software: Es una aplicación informática es la parte lógica del ordenador, es el conjunto de programas que puede ejecutar el hardware para la realización de las tareas de computación a las que se destina.

GD: Librería para generar gráficas dinámicas, su funcionamiento depende por completo del lenguaje php ya que viene incorporada como una extensión del mismo.