

Universidad de las Ciencias Informáticas

Facultad 10



Sistema de Gestión de Banners. Intranet 2

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

Autores: Yanet Avalo Benamor

Yarisleidy Salazar Torres

Tutor: Ing. William Santana Méndez

Ciudad de la Habana, julio de 2007

"No debemos perder la fé en la humanidad, que es como un océano; no se mancha porque algunas de sus gotas estén sucias."

Mahatma Gandhi

DECLARACIÓN DE AUTORÍA

Declaramos ser autores de la presente tesis y reconocemos a la Universidad de las Ciencias Informáticas los derechos patrimoniales de la misma, con carácter exclusivo.

Para que así conste firmamos la presente a los ____ días del mes de _____ del año ____2007____.

Yanet Avalo Benamor

Yarisleidy Salazar Torres

Firma del Autor

Firma del Autor

William Santana Méndez

Firma del Tutor

Datos de Contacto

Tutor: William Santana Méndez

Correo Electrónico: wsantana@uci.cu

Centro de Trabajo: Universidad de las Ciencias Informáticas (UCI)

Formación Académica: 2001-2006 Ingeniero Informático en la CUJAE.

Conocimientos medios-altos:

- Windows.
- Linux
- Lenguaje Pascal, C#, Java, C++, PHP, HTML, XML.
- Base de datos
- Ingeniería y Gestión de Software
- Gráficos por computadora

Otros datos de Interés:

Tesis de pregrado realizada en la UEB Ómnibus Grandes Transtur SA Ciudad Habana con el tema "Sistema de control de Vouchers".

Agradecimientos

A mi abuela, por su amor, cariño, y toda una vida de constante dedicación.

A mi mamá, por su apoyo, comprensión, y la confianza que siempre depositó en mí.

A mi tía Odalis (mi titi), por su esmero, preocupaciones y dedicar tantos años de su vida a mi formación y crianza.

A mi novio Joaco, por su amor, apoyo y ayuda en la elaboración de este trabajo.

Y demás familiares, que me han apoyado en este momento tan importante.

Yanet Avalo Benamor

A mi mamá, por el apoyo, el cariño y estar siempre a mi lado.

A Barto por comprenderme, ayudarme, y ser incondicional en todos estos años.

A Giovanni, por darme su amor, y comprensión.

A Yanet, mi compañera de tesis, por haber hecho posible la culminación de este proyecto.

Yarisleidy Salazar Torres

A Susel, Sara, Dunia, Yoandy, y en especial a nuestro tutor William, por todo el tiempo dedicado, para aclarar nuestras dudas y revisar la tesis.

A Harlem y Franklin por habernos dedicado también parte de su preciado tiempo.

A nuestros compañeros de grupo, por todos los buenos y malos momentos que hemos pasado juntos en el transcurso de estos cinco años.

Y a todas aquellas personas, que a lo largo de toda nuestra vida estudiantil han contribuido a nuestra formación profesional.

Dedicatoria

A mi mamá, que aunque muchas veces estuvimos un poco lejos, quiero que sepas que has sido mi ejemplo, y un gran apoyo en mi vida.

A quienes me criaron con tanto amor y dedicación: A mi abuela y a mi tía Odalis.

A mi hermanito, y a Maria Karla, por alegrarme la vida y formar parte de ella.

A mi novio Joaco, por ser una persona maravillosa, y especial.

A mi familia en general.

Yanet Avalo Benamor

A mi mamá, por confiar siempre en mí.

A mi hermano, por todo su cariño.

A Giovanni por ocupar un pedazo bien grande en mi corazón

Y a todas las personas que me han ayudado a ser mi sueño realidad.

Yarisleidy Salazar Torres

Resumen

La Intranet de la Universidad de la Ciencias Informáticas es actualmente el sitio Web más visitado por los usuarios del centro, en el se encuentran la mayoría de las informaciones de las diferentes áreas, ya sean en el ámbito docente, cultural, o vínculos hacia otros sitios, pues la dinámica propia de este centro en el que las tecnologías de la información y las comunicaciones tienen un peso inigualable, así lo propicia. Pero la Intranet de la UCI presenta muchos problemas en cuanto a arquitectura, diseño y organización de la información, por lo que se quiere lograr un servicio Web dinámico y flexible, que este a disposición y en función de la comunidad universitaria.

La Intranet utiliza diversos banners, en los que se encuentran los educativos, los de deporte, los activos y otros, estos ayudan a transmitir a los usuarios una información directa, además que algunos cuentan con un enlace sirviendo de promoción a otros sitios.

El trabajo que se presenta consiste en realizar un gestor de banners con el objetivo de lograr una herramienta capaz de llevar a cabo el almacenamiento y control de los banners, teniendo en cuenta la seguridad y acceso para los mismos.

Con esta propuesta los administradores de la Intranet podrán gestionar sus banners y realizar una mejor publicación de estos.

Palabras Claves: Gestión, Banners, Intranet, Publicidad, Propaganda, Anuncio, Sitio Web.

Introducción	1
Capítulo 1: Fundamentación teórica.....	5
1.1 Introducción.....	5
1.2 Objeto de estudio.....	6
1.2.1 Flujo actual de los procesos.....	6
1.2.2 Análisis Crítico de la ejecución de los procesos.....	6
1.3 Procesos objeto de automatización.....	7
1.4 Sistemas automatizados existentes vinculados al campo de acción.....	7
1.4.1 Banner Servicio.....	7
1.4.2 Bannerlandia. (Anónimo, 2004).....	7
1.4.3 Galicia City.....	8
1.4.4 Arquitectura en línea.....	8
1.4.5 Sistema de explosión de banners.....	9
1.4.6 ¿Por qué se decidió hacer un sistema automatizado de banners?	9
1.5 Fundamentación de los objetivos.....	9
1.6 Tendencias y Tecnologías Actuales.....	10
1.6.1 ¿Qué es una aplicación Web? Ventajas y desventajas.....	10
1.6.2 Lenguajes de programación Web.....	11
1.6.2.1 Propuesta del Lenguaje de programación.....	15
1.6.3 Sistemas Gestores de Bases de Datos (SGBD).....	16
1.6.3.1 Propuesta del Gestor de Base de Datos.....	19
1.6.4 Entorno de programación.....	19
1.6.4.1 Propuesta del Entorno de Programación.....	21
1.7 Metodologías de desarrollo de software.....	22
1.7.1 Metodologías de desarrollo Rational Unified Process (RUP).....	22
1.7.2 Extreme Programing (XP).....	23
1.7.3 Microsoft Solution Framework (MSF).....	24
1.7.4 Propuesta de la Metodología de desarrollo de software.....	24
1.8 Conclusiones.....	24
Capítulo 2: Modelo del Negocio.....	25
2.1 Introducción.....	25
2.2 Actores y trabajadores del negocio.....	25
2.3 Diagrama de Casos de Uso del Negocio.....	26
2.4 Descripción de los Casos de Uso del Negocio.....	26
2.4.1 CUN Realizar Solicitud de Banners.....	26
2.5 Modelo de Objetos del Negocio.....	29
2.6 Flujo de Requisitos.....	29
2.6.1 Requisitos Funcionales.....	29
2.6.2 Requisitos No Funcionales.....	32
2.7 Conclusiones.....	33
Capítulo 3: Características del Sistema.....	34
3.1 Introducción.....	34
3.2 Actores del sistema a automatizar.....	34
3.3 Casos de Usos del Sistema.....	34
3.4 Diagrama de Casos de Uso del Sistema.....	36
3.4 Descripción de los Casos de Uso del Sistema.....	36
3.6 Conclusiones.....	45
Capítulo 4: Análisis y Diseño del Sistema.....	46
4.1 Introducción.....	46
4.2 Análisis.....	46
4.2.1 Diagramas de Clases de Análisis.....	46
4.2.2 Diagramas de Interacción.....	50

Índice

4.3 Diagramas de Clases del Diseño Web.....	55
4.4 Principios del diseño.....	67
4.4.1 Interfaz de usuario.....	67
4.5 Tratamiento de errores.....	67
4.6 Diseño de la Base de Datos.....	69
4.6.1 Diagrama de clases persistentes de la base de datos.....	69
4.6.2 Diagrama entidad relación de la base de datos.....	70
4.6.3 Descripción de las tablas de la Base de Datos.....	71
4. 7 Conclusiones.....	74
Capítulo 5: Implementación y Prueba.....	75
5.1 Introducción.....	75
5.2 Diagrama de despliegue.....	75
5.3 Diagramas de componentes.....	76
5.4 Modelo de prueba.....	78
5.5 Conclusiones.....	87
Conclusiones.....	88
Recomendaciones.....	89
Referencias Bibliográficas	
Bibliografía	
Glosario de Términos	

Introducción

La publicidad como medio de difusión se remonta a épocas antiguas pero en la actualidad su auge e importancia cobran mayor influencia en el hombre, gracias a los medios de comunicación. Hoy en día se pueden ver y oír divulgación en todas partes, al viajar en automóvil se pueden ver vallas publicitarias y a la vez escuchar en la radio propagandas, al leer un periódico o revista se encuentran anuncios publicitarios, mientras se navega por Internet se observan banners, anuncios y hasta se pueden hacer visitas virtuales. (Roco, 2005)

Esta posee una poderosa fuerza de persuasión, modeladora de actitudes y comportamientos en el mundo de hoy, muy unida a la historia, ya que está presente como resultado de la evolución económica, comercial, técnica y financiera e, inevitablemente, a lo que va ocurriendo en lo social, ideológico y cultural. Puede jugar un importante papel en el proceso por el cual un sistema económico, guiado por normas morales y una sensibilidad hacia el bien común, contribuye al desarrollo humano. Es un instrumento útil para apoyar honesta y éticamente una responsable competitividad que contribuya al crecimiento económico y al servicio del auténtico desarrollo humano. Internet, es actualmente uno de los medios de difusión que permite llegar de forma más precisa a grupos poblacionales específicos, es un medio rápido e interactivo, permite continuas modificaciones y adaptaciones. (Roco, 2005)

En Internet existen muchos anuncios, en los que se encuentran los banners, se les llama así a los anuncios o carteles que se insertan en las páginas Web, que con pulsarlos puede uno dirigirse hacia el sitio del anunciante.

En los sitios Web es de suma importancia utilizar formatos publicitarios como los banners, para que en sus páginas existan anuncios dinámicos o estáticos capaces de transmitirles a los usuarios una información directa, esta forma de divulgación online consiste en incluir una pieza publicitaria dentro de una página Web.

Los banners están diseñados con la intención de atraer la atención, resultar notorios y comunicar el mensaje deseado, además que pueden contar con un vínculo. Este elemento no es solo significativo en Internet, sino también en cualquier sitio Web de nuestro país o el resto del mundo, por la actividad que brindan, dinamismo, colorido y pueden servir de enlaces a otras páginas de interés, su éxito radica en lo llamativo que sea y lo directa que se encuentre la información.

Los banners se crean a partir de imágenes (GIF o JPEG), o de animaciones creadas a partir de tecnologías como Java, Adobe Shockwave y, fundamentalmente Flash, incluso existen los banners flotantes, que son unos banners que se van moviendo por toda la página, el resto se suele colocar en la parte superior e inferior, y a izquierda y derecha de la página. Todo tipo de sitios Web son

susceptibles de incluir toda clase de banners y otros formatos publicitarios.

Actualmente la Intranet de la UCI realiza de forma manual la publicidad de los banners por los administradores de la Intranet, ellos son los encargados de poner en la Intranet los banners que le brinda la dirección de informatización, este problema les dificulta mucho a los administradores de la Intranet la publicación de estos, ya que desconocen ciertas informaciones sobre los banners que exponen al usuario en la misma, quien lo solicitó, qué tipo de banners es, cómo está clasificado, dónde se desea publicar, cuándo, por cuánto tiempo, y si está disponible el mismo para que sea publicado en la Intranet.

El hecho de que toda la actualización mencionada con anterioridad, se realice de forma manual, trae consigo que los banners educativos se han mantenido por dos cursos consecutivos y los banners promocionales no están enlazados con sus respectivos sitios, por lo que la necesidad de una solución informática se requiere.

De lo analizado anteriormente nos planteamos como **problema científico**:

¿Cómo implementar un gestor de banners que les facilite a los administradores de la Intranet de la Universidad de las Ciencias Informáticas una mejor publicidad de los mismos?

El **objeto de estudio** se centra en los sistemas de gestión de banners de las Intranets.

Con las características expresadas anteriormente se refiere como **campo de acción** al sistema automatizado de banners en la Intranet de la Universidad de las Ciencias Informáticas.

Para dar solución al problema antes mencionado se plantea la siguiente **hipótesis**: Si se logra la implementación de un sistema automatizado de banners en la Universidad de las Ciencias Informáticas entonces se obtendrá un mejor funcionamiento en la gestión de los banners, controlando la seguridad y acceso a los banners, y el almacenamiento de toda la información de estos.

Como **objetivo general** se propone implementar un gestor de banners que les proporcione a los administradores de la Intranet de la Universidad de las Ciencias Informáticas la posibilidad de guardar la información de los banners de forma automatizada y manipularlos de la forma más conveniente.

Los **objetivos específicos** que nos trazamos son:

1. Diseñar una Base de Datos debidamente normalizada para almacenar los banners
2. Implementar en el sistema las funcionalidades necesarias para realizar acciones sobre los

banners tales como clasificar, guardar, visualizar y eliminar.

3. Controlar la seguridad del sistema de banners.

Con el objetivo de guiar, controlar y evaluar la investigación se definieron las siguientes **tareas**:

1. Estudio preliminar de la situación actual de la publicación de los banners de la intranet de la Universidad de las Ciencias Informáticas.
2. Estudio y análisis comparativo de los lenguajes de programación relacionados con aplicaciones Web, gestores de bases de datos y sistemas de banners utilizados en Cuba y en el mundo.
3. Análisis y diseño del sistema a desarrollar para la automatización de los banners del módulo de los administradores.
4. Implementación y prueba del sistema a desarrollar para la automatización de los banners del módulo de los administradores.

Para obtener toda la información necesaria que permita utilizarán los siguientes **métodos teóricos y empíricos**:

1. La entrevista: para la recopilación de toda la información necesaria para el diseño de la aplicación.
2. La revisión de documentos: para conocer que es un banners, como se clasifican, la forma en que se almacenan, etc.
3. La modelación: para estudiar las relaciones y cualidades del objeto de estudio.

Con el desarrollo del sistema de banners se espera, que se logre un mejor funcionamiento de los banners, el incremento de la calidad en la publicación de los banners, así como, ayudar a la organización y almacenamiento de las solicitudes a la hora de realizar una publicación.

Estructuración del contenido

En el primer Capítulo se expone una fundamentación teórica. Se redacta la descripción detallada del objeto de estudio de la investigación, los procesos objeto de automatización, se plantean los objetivos generales y específicos del trabajo. Se hace una comparación de las tendencias y tecnologías actuales, así como de los diferentes sistemas existentes.

Introducción

En el segundo Capítulo se refiere a la modelación del negocio. Se describen los casos de uso del negocio, los actores y trabajadores del mismo, así como el modelo de objetos. Se plantean los requisitos funcionales y no funcionales.

En el tercer Capítulo está relacionado con el análisis y descripción del sistema, se determinan los casos de uso del sistema y su descripción. Se definen los actores del sistema.

En el cuarto Capítulo se realiza el Análisis y Diseño. Se documentan las etapas de análisis y diseño según la metodología RUP.

En el quinto Capítulo se realiza la implementación y prueba. Se elaboran los diagramas de despliegue y componentes, y las pruebas del sistema por caso de uso.

Capítulo 1: Fundamentación teórica.

1.1 Introducción.

En muchas ocasiones en que exploramos el mundo de Internet, o una simple Intranet de una universidad, nos encontramos en sus páginas espacios publicitarios, o lo que es lo mismo, anuncios dinámicos o estáticos ubicados en la parte superior, central o derecha en las páginas Web y que logran de cierta manera captar nuestra atención.

Se le conoce con el nombre de Banners a este medio de publicidad más explotado en la Web, no son más que imágenes animadas que se utilizan para captar la atención del navegante y transmitirle alguna información o anuncio. Una vez que el usuario haga un click sobre el mismo son transportados de manera virtual al sitio del anunciante.

Depende de la calidad del banner el resultado que se quiera obtener. En varias ocasiones se ha dicho que los banner no son efectivos y su funcionamiento no es bueno, pero la razón es porque los banner no llevan la información precisa o no están hechos de manera que capten la atención al cliente. Puede que el banner no este correcto porque no contiene la información que describía. Si no se es capaz de realizar un banner que finalmente cumpla con los objetivos propuestos, entonces no tendrá efectividad ninguna.

El formato clásico de banner es horizontal y mide 468x60 píxeles, aunque existen muchos otros formatos en función del soporte (sitio Web que los acoge).

Las tecnologías utilizadas que determinan sus características son:

Formularios: Anuncios que permiten incluir elementos interactivos disponibles en HTML, como cajas de textos y menús desplegados.

Animaciones: Anuncios que utilizan GIF's que permiten crear animaciones simples.

Imágenes: Anuncios estáticos que no usan interactividad o animaciones.

El precio de publicación de los banners es definido por los administradores de cada página de acuerdo al tráfico de la misma, al tamaño, al movimiento, al tiempo y a la cantidad de información.

1.2 Objeto de estudio.

1.2.1 Flujo actual de los procesos.

Los banners son anuncios publicitarios que brindan en la Intranet de la Universidad de las Ciencias Informáticas una información directa de algún tema en específico o de algunas de las áreas y sitios de trabajo que tiene la escuela, como por ejemplo, la biblioteca, el sitio de la FEU, UJC, docencia, etc.; también sirven de vínculo hacia otra página o sitio. Los banners son manipulados por una sola persona, en este caso el administrador de la Intranet o los periodistas, esta persona se encarga de guardarlos en una máquina, insertarlos en la Intranet los días correspondientes a su salida, y eliminarlos cuando estén fuera de fecha, además de tener que modificar un script manualmente para esta "publicación". Los interesados en publicar algún anuncio no pueden hacerlo online, solo esta autorizado el administrador de la Intranet o los periodistas, cuando una individuo solicita un banner para su publicación debe escribirle un correo a los encargados que pertenecen a la dirección de la informatización para que ellos lo aprueben, si lo aprueban y cumple con los siguientes requisitos, entonces el banner solicitado se publica. Los requisitos son:

- Para publicar un flash en la Intranet no debe tener sonidos de fondo.
- Se exige una buena ortografía y que posea toda la información de interés (lugar, hora, fecha, etc.).
- Las dimensiones que debe cumplir son: el superior –horizontales- con ancho 435 y alto 56 y los laterales –verticales- con ancho 125 y alto 250.

1.2.2 Análisis Crítico de la ejecución de los procesos.

La actualización y publicación de los banners que se menciona con anterioridad se realiza de forma manual, lo que provoca que los banners se conserven por tanto tiempo en la Intranet sin ser renovados, esto trae consigo que los usuarios que accedan a la Intranet de la Universidad pierdan el interés por la información que ahí aparece al ser siempre los mismos; además los banners se encuentran guardados en una carpeta de una pc sin seguridad donde cualquier persona no autorizada puede acceder a la misma. Los banners no se encuentran clasificados por categoría lo que se le dificulta el trabajo a los administradores, ya que los mismos pueden ser publicados en una página que no se relaciona con la información que brinda el banner.

1.3 Procesos objeto de automatización.

Dada la existencia de todos los problemas planteados y teniendo en cuenta el alto grado de interés que presentan la publicidad de los banner se hace necesaria la implementación de este sistema de banners que permitirá automatizar diferentes procesos.

Se desea automatizar los banners por categoría, y con ello toda la información del mismo, como nombre del banner, estado, etc. Automatizar que se pueda visualizar, insertar y eliminar un banner cuando este caduque. Automatizar la autenticación de los usuarios dándoles acceso a los servicios que el sistema brinda según sus permisos. Automatizar que nos muestre las categorías existentes e insertar nuevas categorías. Automatizar que se pueda eliminar o agregar un nuevo usuario al sistema. Automatizar la planificación de los banners, la fecha que se publicarán y se retirarán del sitio, la frecuencia, y el tipo de planificación (Diariamente, Semanalmente, Mensualmente).

1.4 Sistemas automatizados existentes vinculados al campo de acción.

En la actualidad existen infinidad de sistemas de intercambio en la red. A continuación se mostrarán los Sistemas de intercambios más usados por los administradores de red:

1.4.1 Banner Servicio.

Es un servicio gratuito cuyo objetivo es ayudar a conseguir más tráfico en las páginas Web. Cada miembro puede elegir su formato de banner. Aceptan GIF, JPG, GIF's animados y ShockWave. El tamaño de banner ha de ser 468x60 píxeles, que es el tamaño más óptimo para su mejor visualización y la posibilidad de conseguir un buen diseño. Con el sistema el usuario controla de forma permanentemente el estado de su cuenta, y sabe cuantos créditos disponibles tiene, cuantos banners se publicaron, que banners publicó el usuario y la aceptación que tuvieron. Brinda la posibilidad de cambiar el gráfico en cualquier momento, para así el cliente pueda en todo momento optar por poner otros banners que tengan más aceptación. (Anónimo, 2004)

1.4.2 Bannerlandia. (Anónimo, 2004)

Es un intercambio de banners con más de tres años en línea y miles de usuarios que permite promocionar el sitio gratuitamente.

Características principales del programa de intercambio:

- Intercambio 2 a 1: por cada 2 banners que muestre el cliente en su sitio se mostrará un banner suyo en otro sitio. La diferencia permite que ellos comercialicen banners para poder mantener el

servicio gratuito.

- El usuario puede tener hasta 20 banners distintos y publicar banners en cuantas páginas quiera.
- Aceptan banners en formato GIF de 468x60, con o sin animaciones, hasta 15K de peso. El código del sistema está preparado para que próximamente acepten banners Flash y HTML.
- Generador de banners. Si el usuario no tiene banners puede generarlos en menos de 1 minuto con una herramienta que posee el sistema.
- Control de contenido. El cliente puede definir que categorías de sitios desea promover y cuales no quiere promocionar.
- Se puede seleccionar donde mostrar la publicidad: por categorías o idioma del usuario.
- Numerosos informes acerca de los banners del sitio y las visitas.

1.4.3 Galicia City.

Galicia City es un servicio de intercambio de banner que se financia con la ayuda de sus patrocinadores y los anuncios que los mismos hagan serán entonces los banner expuestos en sus páginas. Pueden participar usuarios particulares, empresas, asociaciones, instituciones privadas o públicas que tengan páginas Web alojadas en algún servidor conectado a Internet.

Es el primer servicio de este tipo que surge en la Comunidad Gallega con el propósito de que sus usuarios puedan mostrar en sus propias páginas, banners de otros usuarios, al tiempo que estos hacen lo mismo con su banner.

Cuando el usuario visita la Web, y cargan cada uno de los banners, el patrocinador recibe medio crédito. Cada crédito le da derecho a una impresión de su banner en páginas de otros miembros.

Los participantes en este sistema deben colocar el código HTML facilitado por Galicia City - Intercambio de banners, en un lugar visible de su página, donde no se deprecie o desvalore el contenido del banner, también registra las direcciones IP de cada "hit" recibido, lo que permite detectar accesos automáticos/repetitivos, es decir, que los miembros no podrán "inflar" artificialmente su cuenta, a través de programas o cualquier otro método que genere "hits" automática o continuamente en sus páginas. (Rodríguez, 2002)

Este sistema no admite banners de anunciantes, cuyas páginas no considere apropiadas u oportunas. No se aceptan altas de banners o Web considerados ofensivos, de carácter pornográfico, racista, terrorista o cualquier otro contenido impropio, y no se hace responsable de los daños o perjuicios que la utilización de mismo pueda ocasionar.

1.4.4 Arquitectura en línea.

Es un emprendimiento digital dedicado al asesoramiento y publicación especializada en arquitectura, arte y diseño dentro de Internet. Para la publicidad de su empresa ha implementado un sistema de

banner rotativos, con enlaces a sus propias páginas. Con solo enviar un banner y un texto que cumpla con las condiciones impuestas por ellos, aparecerán de forma aleatoria en las páginas visitadas diariamente en sus Web.

Este sistema le asigna un usuario y contraseña una vez contratado el servicio. Al ingresar al sistema se puede visualizar las estadísticas relacionadas con sus banners.

1.4.5 Sistema de explosión de banners.

Este sistema ofrece servicio de intercambio de banner, es totalmente gratis. Para su funcionamiento primero hay que guardar un documento que ellos te ofrecen y utilizando un navegador (*browser*) se guarda con la extensión *explosion.htm*. Después de eso te dan una serie de requisitos a tener en cuenta hasta finalmente colocar el banner en la Web.

1.4.6 ¿Por qué se decidió hacer un sistema automatizado de banners?

Se decidió realizar la implementación de un sistema automatizado de banners en la Universidad de las Ciencias Informáticas para la Intranet de la misma, y no darle uso a uno ya elaborado, debido a que el sistema de gestión de banners que se implementará presentará condiciones específicas que necesita la Intranet de la Universidad. Además este sistema asegurará el acceso y el control de sus servicios, ya que tendrá una base de datos donde van a estar registrados el personal que va a interactuar con el mismo.

Además, al hacer uso de otros sistemas, incluiría gastos monetarios innecesarios para una potencia tan grande como es nuestro país, capaz de suplir sus propias necesidades a la hora de demostrar sus capacidades.

Con la realización de este proyecto para la Intranet de la UCI, no solo nos crea una independencia en el desarrollo de sistemas de gestión, sino que también se pone en práctica los conocimientos adquiridos durante estos años en la universidad.

1.5 Fundamentación de los objetivos.

Este trabajo tiene como objetivo implementar un gestor de banners que les proporcione a los administradores de la Intranet de la Universidad de las Ciencias Informáticas la posibilidad de guardar la información de los banners de forma automatizada y manipularlos de la forma más conveniente.

Los objetivos específicos que nos trazamos son:

1. Diseñar una Base de Datos debidamente normalizada para almacenar los banners
2. Implementar en el sistema las funcionalidades necesarias para realizar acciones sobre los banners tales como clasificar, guardar, visualizar y eliminar.
3. Controlar la seguridad del sistema de banners.

Con estos objetivos trazados se lograría un gestor de banners capaz de suplir las necesidades de los administradores de la intranet y un mejor servicio de publicación de banners, ya que con el diseño de la Base de Datos se almacenarán los banners con toda la información que el administrador necesita a la hora de su publicación (nombre, estado, categoría, y el URL del sitio donde se van a publicar), además no solo guardaría la información del banner, sino de los usuarios que estarán registrados en el sitio. La implementación del sistema conseguiría que acciones que el administrador realiza actualmente de forma manual, se ejecuten de forma automática, logrando una mejor planificación de los banners, los cuales no se conservarían tanto tiempo en el sitio sin ser renovados, También se implementaría un control en la seguridad de los banners, automatizando el acceso de cada usuario así como el control de privilegios de cada usuario una vez dentro del sistema, logrando que cada cual realice dentro del sistema solo lo que le esta permitido..

1.6 Tendencias y Tecnologías Actuales.

En el presente epígrafe se hará un análisis y un estudio comparativo de las tecnologías que pueden ser de utilidad para el mejor entendimiento del sistema, además se concebirá una propuesta de las diferentes herramientas que se utilizarán en el posterior desarrollo de la aplicación.

1.6.1 ¿Qué es una aplicación Web? Ventajas y desventajas.

Una aplicación Web es un programa de computadora que los usuarios utilizan accediendo a un servidor Web a través de Internet o de una Intranet. Las aplicaciones Web utilizan las tecnologías existentes para generar contenidos dinámicos y permitir a los usuarios del sistema modificar la lógica del negocio en el servidor.

Esta aplicación informática es popular debido a la practicidad del navegador Web como cliente ligero.

Ventajas principales de las aplicaciones Web:

- Es un sistema multiplataforma, ya que se puede migrar de sistema y cambiar el hardware libremente, y no afecta el funcionamiento de las aplicaciones de servidor.
- Se necesita solo de un computador con un buen navegador Web para el uso de esta aplicación, ya que no se requieren complicadas combinaciones de hardware/software.

- El usuario no se da cuenta de las actualizaciones del software ya que estas se realizan automáticamente.
- Al funcionar en un navegador, para el uso de una aplicación Web, se requiere solo de un conocimiento básico de la informática.
- Proporciona facilidad en el trabajo a distancia, y se puede trabajar con conexión a Internet desde cualquier PC o computador portátil.

Desventajas de las aplicaciones Web:

- Difícil implementación de la seguridad.
- Es necesaria una conexión rápida y permanente.
- El desempeño puede ser bajo en un servidor potente debido a que se están enviando los datos y el diseño ya que se pide un documento HTML.
- Existen limitaciones en los documentos estándar HTML.

1.6.2 Lenguajes de programación Web.

Existen numerosos lenguajes de programación utilizados para el desarrollo de Aplicaciones Web. Estos lenguajes se dividen en lenguajes de lado servidor (son aquellos lenguajes que son reconocidos, ejecutados e interpretados por el propio servidor). Dentro de los más importantes se encuentra PHP, ASP, PERL y JSP. Los lenguajes de lado cliente es el otro lenguaje y son aquellos que pueden ser directamente "digeridos" por el navegador, como son HTML, Java y JavaScript, los cuales son simplemente incluidos en el código HTML.

ASP.

ASP (Active Server Pages - Página Activa en el Servidor) es una tecnología del lado servidor desarrollado por Microsoft para páginas Web generadas dinámicamente.

La tecnología ASP está estrechamente relacionada con el modelo tecnológico de su fabricante. Intenta ser solución para un modelo de programación rápida ya que programar en ASP es como programar en VisualBasic, pero con muchas limitaciones.

ASP no es en sí mismo un lenguaje de programación, si no más bien un marco sobre el que se construyen aplicaciones basadas en Internet, apoyándose para ello en el lenguaje HTML, en lenguajes de script conocidos (generalmente VBScript, pero también JavaScript-Jscript para Microsoft-, Perl, y otros.), en motores de bases de datos y en el lenguaje de consulta SQL. ASP ha pasado por cuatro iteraciones mayores, ASP 1.0 (distribuido con IIS 3.0), ASP 2.0 (distribuido con IIS

4.0), ASP 3.0 (distribuido con IIS 5.0) y ASP.NET (parte de la plataforma .NET de Microsoft). Las versiones PRE-.NET se denominan actualmente (desde 2002) como ASP *clásico*. (Anónimo, 2004) Evidentemente una de las desventajas es que su empleo se realiza solo sobre plataformas funcionando bajo sistema Windows NT.

ASP.NET

ASP.NET constituye un conjunto de tecnologías de desarrollo de aplicaciones Web (es la compañía sucesora de ASP) y es comercializada por la compañía Microsoft (forma parte de la plataforma .NET de Microsoft). Su uso por los programadores es para la construcción de sitios Web domésticos, aplicaciones Web y otros servicios como XML. Esta tecnología llamada Active Server Pages fue introducida por Microsoft en 1996, por lo que se puede afirmar que no es nueva. Es una tecnología de páginas activas que nos permite hacer uso de diferentes scripts y componentes con el tradicional HTML. Para mostrar páginas generadas dinámicamente. (Pey, 2001)

En el modelo de desarrollo Web basado en páginas activas, la programación ASP actual tiene diversas limitaciones:

- Para que todo ocurra en una página Web, es habitual escribir una gran cantidad de código para resolver necesidades sencillas. ASP.NET incorpora un modelo declarativo a la programación Web: los controles de servidor funcionan en una página Web simplemente declarándolos.
- ASP es un tanto desorganizado. En una página ASP podemos incluir casi todo: HTML plano, código de scripting, objetos COM y texto. No hay una distinción formal entre el contenido de una página y su comportamiento: simplemente, insertamos código en la página, y a ver qué pasa. ASP.NET impone un cierto orden sobre el modelo de programación estándar ASP.
- La tercera limitación en el desarrollo con ASP es que con el tradicional utilizamos lenguajes de scripting no tipados. Podemos instalar otros motores de scripting que impongan verificación de tipos; sin embargo, no son universalmente conocidos o utilizados como los anteriores. ASP.NET claramente separa la porción basada en script de una página Web de su contenido.

PERL.

PERL, Lenguaje Práctico para la Extracción e Informe, es un lenguaje especializado en el procesamiento de textos, particularmente, extraer y validar las respuestas a cuestionarios incluidos en páginas Web.

Perl es un lenguaje interpretado que tiene muchas utilidades, pero está orientado principalmente a la

búsqueda, extracción y formateo de ficheros de tipo texto. También es muy usado para el manejo y gestión de procesos (estado de procesos, conteo y extracción de parámetros característicos, etc.)

Aunque desarrollado originalmente en un entorno UNIX, actualmente hay versiones para casi todos los sistemas operativos: Windows, Mac OS, de manera que es un lenguaje multiplataforma. Es gratuito, libre y uno de los lenguajes más utilizados en la programación de CGI script para el intercambio de información entre aplicaciones externas y servicios de información. (Anónimo, 2007)

Tiene la desventaja de ser un lenguaje que consume muchos recursos de la máquina, lo que significa que no es muy ligero. Si no se escribe con cuidado puede llegar a ser un código ilegible y no se pueden compilar programas con Perl.

JSP.

JavaServer Pages (JSP – Páginas de Servidor Java) es la tecnología para generar páginas Web de forma dinámica en el servidor, desarrollado por Sun Microsystems basado en script que utilizan una variante del lenguaje Java. La tecnología JSP es una tecnología Java que permite a los programadores generar dinámicamente HTML, XML o algún otro tipo de página Web. Esta tecnología permite al código Java y a algunas acciones predefinidas ser embebidas en el contenido estático. En las JSP se escribe el texto que va a ser devuelto en la salida (normalmente en código HTML) incluyendo código Java dentro de él para poder modificar o generar contenido dinámicamente. (Arambillette, 2002)

La principal ventaja de JSP frente a otros lenguajes es que permite integrarse con clases Java (.class) lo que permite separar en niveles las aplicaciones Web, almacenando en clases Java las partes que consumen más recursos así como las que requieren más seguridad, y dejando la parte encargada de formatear el documento HTML en el archivo JSP.

Sin embargo JSP no se puede considerar un script al 100% ya que antes de ejecutarse el servidor Web compila el script y genera un servlet, por lo tanto se puede decir que aunque este proceso sea transparente para el programador no deja de ser una aplicación compilada. La ventaja de esto es, que tiene algo más de rapidez y dispone del API de Java en su totalidad.

Java Script.

JavaScript es un lenguaje de programación interpretado del lado del cliente porque es el navegador el que soporta la carga de procesamiento. Únicamente depende del cliente y no del sistema operativo, sólo necesita un browser capaz de interpretarlo. Gracias a su compatibilidad con la mayoría de los navegadores modernos, es el lenguaje de programación del lado del cliente más utilizado. (Anónimo, 2006)

Cualquier persona puede desarrollar aplicaciones escritas en JavaScript del mismo modo que realiza

páginas HTML.

Por otro lado, JavaScript no es un lenguaje válido para desarrollar aplicaciones concurrentes o de acceso compartido.

Está diseñado para controlar la apariencia y manipular los eventos dentro de la ventana del navegador Web. Este se integra directamente en páginas HTML (como programas propiamente dichos, combinando funciones y sentencias, e introduciendo manejadores de eventos JavaScript en etiquetas HTML) y la ventaja que presenta sobre el HTML es que permite crear páginas Web más dinámicas, lo que las hace más atractivas para el usuario.

XSLT.

XSLT (Extensible Stylesheet Language Transformation, en castellano algo así como lenguaje de transformación basado en hojas de estilo), es un lenguaje que se usa para convertir un documento XML en otro documento XML e incluso en otros formatos.

XSLT se basa en XPath para realizar la búsqueda de información a través del documento XML. El proceso de transformación se basa en plantillas, unidas al documento fuente a transformar. Dichas plantillas identifican una estructura a partir de la cual realizar la transformación (con XPath), así como las acciones a realizar con dicha estructura: recorrerla, obtener el dato de la etiqueta, el valor de alguno de sus atributos, contar cuantos elementos tiene la etiqueta anidada, etc. Esas reglas de plantilla alimentan a un procesador de XSLT, el cual realiza las transformaciones deseadas colocando el resultado en un archivo de salida. (Flores, 2006)

Actualmente XSLT es muy usado en la edición de páginas Web, generando páginas HTML o XHTML. La unión de XML y XSLT permite separar el contenido de la presentación, aumentando así la productividad.

PHP. (S., 2001)

PHP (Personal Home Page) es el acrónimo recursivo de PHP: Hypertext Preprocessor.

PHP es un lenguaje de programación usado generalmente para la creación de contenidos para sitios Web.

Este lenguaje se acopla al HTML (páginas web) para definir procedimientos que ha de realizar el servidor de Web, por ejemplo procesar un formulario, enviar o extraer datos de una base de datos (acoplándose también con un lenguaje de tipo SQL), enviar una u otra página Web según determinadas condiciones prefijadas por el programador, etc.

Últimamente también se usa para la creación de otro tipo de programas incluyendo aplicaciones con interfaz gráfica usando la biblioteca GTK+.

PHP fue originalmente diseñado en Perl, seguidos por la escritura de un grupo de CGI binarios

escritos en el lenguaje C por programador Danés – Canadiense Rasmus Lerdorf en el año 1994, pero como PHP está desarrollado en política de código abierto muchos desarrolladores han hecho sus contribuciones, siendo la PHP 5.2.3 la versión más reciente. Esta versión incluye PDO (Objetos de Información de PHP o PHP Data Objects) y mejoras utilizando las ventajas que provee el nuevo Zend Engine 2.

PHP es gratuito e independiente de la plataforma, pues existe un módulo de PHP para casi cualquier servidor Web. Esto hace que cualquier sistema pueda ser compatible con el lenguaje, lo que significa una ventaja importante, ya que permite portar el sitio desarrollado en PHP de un sistema a otro sin prácticamente ningún trabajo. Además, es seguro, con gran librería de funciones y mucha documentación.

Es un lenguaje multiplataforma que 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.

La característica más potente de PHP es su soporte para una gran cantidad de bases de datos que se utilizan actualmente, destacando su conectividad con MySQL.

1.6.2.1 Propuesta del Lenguaje de programación.

Se utilizará PHP como lenguaje de programación para la implementación del sistema, debido a que es un lenguaje "open source" interpretado de alto nivel embebido en páginas HTML, y su código es ejecutado en el servidor. Este lenguaje nos permite no solo leer, sino también manipular datos desde diversas fuentes, incluso los datos que pueden ser ingresados por los usuarios desde formularios HTML.

El código PHP es mucho más legible que el de PERL, este último lenguaje consume muchos recursos de la máquina haciéndolo menos ligero.

PHP no es el único lenguaje de programación cliente/servidor Web que existe, entre las alternativas más conocidas esta ASP por parte de Microsoft, el cual corre bajo el servidor Web propietario de Microsoft: IIS, debido a esto es de pago, por lo que es muy poco usado y aunque es muy potente su metodología de programación es poco robusta y no logra superar a PHP como lenguaje de programación Web.

JSP por parte de Java, es la otra alternativa desarrollada por Sun Microsystems, este se suele usar más para scripts de línea de comando, mientras que PHP se usa más para páginas dinámicas. PHP a la hora de servir páginas Web es más eficaz que JSP, ya que se integra como módulo en Apache, y así se evita la carga del intérprete en memoria cada vez que se ejecuta una página. PHP al ser una herencia positiva de Perl facilita al programador muchas cosas que en JSP son más difíciles de programar. Por ejemplo, el acceso a bases de datos es mucho más sencillo en PHP, ya que el API de MySQL, por ejemplo, viene integrado en el propio lenguaje.

1.6.3 Sistemas Gestores de Bases de Datos (SGBD).

Los Sistemas Gestores de Bases de Datos (SGBD) son un tipo de software muy específico, dedicado a servir de interfaz entre las bases de datos y las aplicaciones que la utilizan. Se compone de un lenguaje de definición de datos, de un lenguaje de manipulación de datos y de un lenguaje de consulta.

Los SGBD deben proporcionar a los usuarios la capacidad de almacenar datos, acceder a ello y actualizarlos, de forma rápida, estructurada y ocultándole la estructura física interna (la organización de los ficheros y las estructuras de almacenamiento).

Existen distintos objetivos que deben cumplir los SGBD:

- Abstracción de la información.
- Redundancia mínima.
- Consistencia.
- Seguridad.
- Integridad.
- Tiempo de respuesta mínimo.

PostgreSql. (Castellanos, 2005)

PostgreSql es un sistema de administración de bases de datos relacional libre. Tiene más de 15 años de desarrollo activo y se ha ganado la reputación de ser confiable. Corre en la mayoría de los Sistemas Operativos más utilizados incluyendo, Linux, varias versiones de UNIX, BeOS y Windows. Puedes crear varios tipos de datos y hacer herencias entre objetos, tiene transacciones, integridad referencial, vistas, y multitud de funcionalidades, pero es lento y pesado.

PostgreSQL se diseñó como una base de datos orientada a objetos, es decir, una ORDBMS. Esto significa, que las tablas no son tablas, sino objetos, y las tuplas son instancias de ese objeto.

Cumple la prueba ACID (Atomicidad, Consistencia, Integridad, Durabilidad) y tiene soporte completo para llaves foráneas, joins, vistas, subconsultas, disparadores (triggers), y procedimientos almacenados (en varios lenguajes). Incluye la mayoría de los tipos de datos de los estándares SQL92 y SQL99. También soporta almacenamiento de objetos grandes (imágenes, sonido y video) Así, como sus propias interfaces de programación para C/C++, Java, Perl, Python, Ruby, Tcl, ODBC, entre otros, y una documentación muy completa (aunque la documentación se encuentra en inglés hay un proyecto de traducción gestándose).

Han incorporado la llamada MVCC (multiversion concurrency control) con lo que los bloqueos de escritura actúan sólo en la sesión del cliente, no en las de los demás clientes.

SQL Server. (Castellanos, 2005)

Microsoft SQL Server es un sistema de gestión de bases de datos relacionales para Windows, basado en el lenguaje SQL y capaz de poner a disposición de muchos usuarios grandes cantidades de datos de manera simultánea. SQL Server es el sistema de gestión de base de datos representativa de la firma mundialmente conocida Microsoft. Es un software bajo licenciamiento comercial.

SQL Server 2000 proporciona agilidad a sus operaciones de análisis y administración de datos al permitir a su organización adaptarse rápida y fácilmente para obtener ventaja competitiva en un entorno de cambios constantes.

Entre sus características figuran:

- Soporte de transacciones: Las interacciones con las estructuras de datos se realizan de una sola vez o no se realizan.
- Gran estabilidad: El nivel de fallos disminuye por debajo de un determinado umbral, que varía dependiendo de la estabilidad que se requiera.
- Gran seguridad.
- Alta Disponibilidad: Permitiendo configuraciones de alta disponibilidad más complejas. Al permitir que otros servidores de 64 bits puedan recuperar la ejecución de aplicaciones críticas en caso de paradas del servicio planificadas o no previstas.
- Escalabilidad: Capacidad del sistema informático de cambiar su tamaño o configuración para adaptarse a las circunstancias cambiantes, sin perder la calidad de los servicios. Permite realizar un escalamiento hasta 32 CPU y 64 gigabytes (GB) de RAM, siendo capaz de manejar al máximo multiprocesamiento simétrico aprovechando al máximo el hardware.
- Soporta procedimientos almacenados: Al ser ejecutado, en respuesta a una petición de usuario, el procedimiento permite deshacerse de la sobrecarga resultante de comunicar grandes cantidades de datos salientes y entrantes.
- Permite trabajar en modo cliente-servidor: La información y datos se alojan en el servidor y los clientes de la red sólo acceden a la información.

Interbase. (Castellanos, 2005)

Es una base de datos que tiene 16 años de experiencia en el sector de las bases de datos comerciales. La mayoría de los gestores de Backup soportan interbase (como ArcServe). En general es una base de datos buena, que incluye gestores remotos a muy alta calidad para Windows (como Maraton).

Interbase posee muchas herramientas de pago, que hacen de este sistema un producto profesional preparado para cualquier proyecto medio que necesite una base de datos realmente fiable.

Interbase tiene la mayoría de funcionalidades de una base de datos comercial: triggers, tratamiento especial de blobs, backup On-line, gran escalabilidad, bases de datos de solo lectura (para ponerlas en CD-Rom), integridad referencial en cascada o el autotuning. Dispone, de soporte directo para PHP.

MySQL. (Castellanos, 2005)

MySQL es uno de los sistemas gestores de bases de datos más populares desarrollados bajo la filosofía de código abierto que nos permite ejecutar sentencias SQL de manera interactiva o por lotes. Es la base de datos más usada y desarrollada actualmente en el mundo, es proporcionado por MySQL AB (es la empresa que proporciona servicios en torno al servidor de bases de datos MySQL). Su origen se basa en la necesidad que había de crear un manejador de bases de datos que fuera todavía más rápido que MS SQL.

Este sistema es usado con mucho éxito en ambientes de mucho rigor y exigencia por varios años. Se encuentra en un constante desarrollo, y ofrece actualmente una gran variabilidad y utilidad de funciones. Es un servidor bastante idóneo para acceder a bases de datos en Internet debido a su velocidad, conectividad y seguridad.

MySQL Server, cuenta con una variedad de tipos de tablas para el almacenamiento de la información, dependiendo de las necesidades y cantidades de información que una organización puede tener, se elige el tipo de tabla con la que más se acople.

Entre las características disponibles de las últimas versiones se puede destacar:

- Disponibilidad para la mayoría de las plataformas y sistemas operativos.
- Su bajo consumo lo hacen apto para ser ejecutado en una máquina con escasos recursos sin ningún problema.
- Diferentes opciones de almacenamiento, si se desea velocidad en las operaciones.
- Transacciones y claves foráneas.

- Conectividad segura.
- Replicación.
- Búsqueda e indexación de campos de textos.
- Velocidad a la hora de realizar las operaciones.

MySQL es una solución franca y rentable que provee:

- Apoyo y soporte a las empresas por parte de los propios profesionales que desarrollan MySQL.
- Modernización Automatizada y alarmas.
- Licencia de open source flexible.
- Garantías.
- Certificado y preparado para implantar en producción.
- Automáticamente recibe actualizaciones del software.

1.6.3.1 Propuesta del Gestor de Base de Datos.

Se utilizará MySQL como gestor de base de datos, primeramente porque está desarrollada bajo la filosofía de código abierto.

A diferencia de PostgreSQL su principal objetivo de diseño fue la velocidad, y consume muy pocos recursos, tanto de CPU como de memoria, mientras que PostgreSQL es lento, pesado, y consume bastante recursos.

Una de las ventajas que tiene MySQL es que a partir de la versión 3.23.19 está siendo desarrollado bajo la licencia GPL, a diferencia de SQL Server que posee como desventaja altos costos de licencia y al no ser multiplataforma solo esta disponible en Sistemas Operativos de Microsoft.

MySQL, también tiene la ventaja de ser un SGBD robusto, que puede almacenar gran cantidad de datos, es rápido, seguro, estable, gratuito y soporta múltiples lenguajes de programación, con lo que puede conectarse a una base de datos de este tipo a través de cualquiera de ellos.

1.6.4 Entorno de programación.

Zend Studio. (Alvarez, 2006)

Un programa orientado a desarrollar aplicaciones web, en lenguaje PHP. El programa, además de

servir de editor de texto para páginas PHP, proporciona una serie de ayudas que pasan desde la creación y gestión de proyectos hasta la depuración de código.

El programa entero está escrito en Java, lo que a veces supone que no funcione tan rápido como otras aplicaciones de uso diario. Sin embargo, esto ha permitido a Zend lanzar con relativa facilidad y rapidez versiones del producto para Windows, Linux y MacOS, aunque el desarrollo de las versiones de este último sistema se retrase un poco más.

La parte del programa que nos permite escribir los scripts es bastante útil para la programación en PHP. La interfaz está compuesta por varias partes, en las que encontramos un explorador de archivos, una ventana de depuración, los menús y otra para mostrar el código de las páginas.

Lo más destacable es que contiene una ayuda contextual con todas las librerías de funciones del lenguaje que asiste en todo momento ofreciendo nombres de las funciones y parámetros que deben recibir. Aunque esta ayuda contextual no solo se queda en las funciones definidas en el lenguaje, sino que también reporta ayudas con las funciones que vayamos creando nosotros, incluso en páginas que tengamos incluidas con la función `include()`.

Otras ayudas que ofrece a la hora de escribir son las típicas en editores avanzados, como permitir editar varios archivos, y moverse fácilmente entre ellos, marcar a qué elementos corresponden los inicios y cierres de las etiquetas, paréntesis o llaves, moverse al principio o al final de una función, identificación automática del código, etc.

Zend Studio dispone de una herramienta muy interesante de debug o depuración. Gracias a ella podemos ejecutar páginas y conocer en todo momento el contenido de las variables de la aplicación y las variables del entorno como las cookies, las recibidas por formulario o en la sesión. Podemos colocar puntos de parada de los scripts y realizar las acciones típicas de depuración.

Además de la ventana para visualizar el contenido de las variables, dispone de otras donde muestra la salida del script según se va generando, y otra donde se pueden ver las alertas y errores. Las posibilidades se completan con distintos tipos de depuración, en local, en remoto o a partir de una URL.

Quanta.

Quanta es un editor web desarrollado para el escritorio KDE. Soporta una multitud de lenguajes como HTML, Javascript, CSS, PHP, SQL, XML, ColdFusion, Perl, DTML, Zope o C++, lo que hace que hoy por hoy sea uno de los editores más utilizados en el desarrollo de páginas web.

Este editor es un híbrido entre un editor web de texto y uno WYSIWYG ya que nos permite tanto hacer la página viendo el resultado final como directamente desde código.

Quanta nos ofrece multitud de funcionalidades, entre ellas, coloreado de sintaxis para todos los lenguajes soportados, cliente FTP integrado para colgar nuestra página de una manera fácil y rápida, cajas de diálogos contextuales, donde podremos elegir las etiquetas HTML que queremos utilizar, excelente navegador de directorios integrado, con el que accederemos a nuestros ficheros

fácilmente, completo panel de previsualización en el que podremos ver nuestro resultado final, validador HTML integrado en la propia aplicación, soporte de extensiones/plugins para añadir funcionalidades extra, y gran integración con el escritorio KDE, al estar desarrollado para él, lo que nos permitirá trabajar más a gusto con la aplicación si este es el escritorio que utilizamos.

Dreamweaver. (Ciberaula 2006)

Dreamweaver fue uno de los primeros editores de HTML. En la actualidad, existen otros editores de páginas Web, pero sin duda es Dreamweaver la aplicación de este tipo más usada en el sector de diseño y programación Web. Posee, como toda la línea Macromedia/Adobe, excelentes funcionalidades e integración con otras herramientas.

La aplicación permite crear sitios de forma totalmente gráfica, y dispone de funciones para acceder al código HTML generado. Permite la conexión a un servidor, a base de datos, soporte para programación en ASP, PHP, Javascript, cliente FTP integrado, etc.

Con Dreamweaver podemos crear páginas HTML sin tener que preocuparnos por el código HTML (que es bastante correcto y da la posibilidad de adaptarlos automáticamente a navegadores más antiguos), recordar todos los "tags" necesarios para componer nuestra página o tener que previsualizar en nuestra cabeza cual será el resultado compositivo del documento final.

Para facilitar aun más el proceso de creación de una página Web, Dreamweaver añade junto a las opciones que permiten formatear un documento HTML otras opciones que pueden ser de gran utilidad (por ejemplo javascripts, behaviors).

Permite mantener y modificar la apariencia completa de un sitio modificando un solo documento. Por otro lado, alrededor de esta herramienta de diseño y composición se han ido añadiendo otras opciones que permiten gestionar un sitio completo, como puede ser el cliente FTP incluido en Dreamweaver. Utiliza la tecnología propia de Macromedia "Roundtrip".

Es un programa que permite ser personalizado en un alto grado. Presenta una tecnología abierta, por lo que, cualquier persona puede personalizar los menús de la aplicación, añadir nuevos comandos, así como eliminar parte de los existentes.

1.6.4.1 Propuesta del Entorno de Programación.

Por todas las ventajas antes mencionadas se escogió Dreamweaver como editor HTML, es una herramienta orientada a crear, editar y gestionar sitios Web enteros.

Zend Studio a pesar de sus ventajas para programar en PHP, es complicado hacerlo correr con MySQL. El depurador pierde eficiencia cuando se trata de usar otras bases de datos que no sean

Oracle. Tampoco se lleva bien con los combo WAMP (Windows Apache MySQL PHP) como EasyPHP, Apache2triad, PHPTriad, FoxServer, AppServer y otros, parece que solamente se lleva bien con Zend Studio Server (otro WAMP).

Quanta es un completo editor HTML, pero le falta mucho aún para llegar a la calidad y flexibilidad de Dreamweaver, por pulir sobre todo en el área de multimedia. Además se hace muy complicado manejar sus herramientas.

Dreamweaver supera en opciones y calidad al resto de sus competidores, ofrece mayor integración con productos conocidos como Macromedia Flash, permite disponer de dos vistas simultáneamente: el código Web y el modo gráfico, y la calidad, presenta extensiones descargables gratuitas desde varias páginas de la red, el manejo sencillo de sus herramientas de edición, hace que el trabajo diario sea más rápido y cómodo, y Mediante el Extensión manager se pueden añadir nuevas características lo que hace a este editor más potente.

.1.7 Metodologías de desarrollo de software.

Las metodologías imponen un proceso disciplinado sobre el desarrollo de software con el fin de hacerlo más predecible y eficiente.

El uso de una metodología para la elaboración de un producto informático, garantiza determinadas características en el mismo, dentro de ellas la calidad, factor clave tanto para el cliente como para el productor.

El tiempo es uno de los factores críticos que afecta todo producto, más si no se ha hecho un buen estudio de la aplicación que se va a realizar. La cantidad de personal; algunas veces en exceso, otras en déficit, los sistemas de organización, los métodos de control, el dominio sobre el tema y sobre las herramientas de desarrollo por parte de los analistas y programadores, la falta de conocimientos sobre asuntos informáticos por el lado de los clientes, son otros de los factores que afectan todo el ciclo de desarrollo de una aplicación.

1.7.1 Metodologías de desarrollo Rational Unified Process (RUP).

El Proceso Unificado Racional o RUP (Racional Unified Process), es un proceso de desarrollo de software y junto con el Lenguaje Unificado de Modelado (UML), constituye la metodología estándar más utilizada para el análisis, implementación y documentación de sistemas.

Como todo proceso de desarrollo, su objetivo es elevar la calidad del software por todas las fases por las que pasa, a través de una mayor transparencia y control sobre el proceso. Es una forma disciplinada de asignar tareas y responsabilidades, que pretende implementar las mejores prácticas en ingeniería de Software. Es iterativo e incremental, se centra en la arquitectura y es guiado por los casos de uso.

Según la metodología RUP el ciclo de vida de un proyecto se divide en las siguientes fases:

- Fase de concepción
- Fase de elaboración
- Fase de construcción
- Fase de transición

El ciclo de vida que se desarrolla por cada iteración, es llevada bajo dos disciplinas:

Disciplina de Desarrollo

Ingeniería de Negocios: Entendiendo las necesidades del negocio.

Requerimientos: Trasladando las necesidades del negocio a un sistema automatizado.

Análisis y Diseño: Trasladando los requerimientos dentro de la arquitectura de software.

Implementación: Creando software que se ajuste a la arquitectura y que tenga el comportamiento deseado.

Pruebas: Asegurándose que el comportamiento requerido es el correcto y que todo lo solicitado esta presente.

Disciplina de Soporte

Configuración y administración del cambio: Guardando todas las versiones del proyecto.

Administrando el proyecto: Administrando horarios y recursos.

Ambiente: Administrando el ambiente de desarrollo.

Distribución: Hacer todo lo necesario para la salida del proyecto.

Una particularidad de esta metodología es que, en cada ciclo de iteración, se hace exigente el uso de artefactos, siendo por este motivo, una de las metodologías más importantes para alcanzar un grado de certificación en el desarrollo del software.

1.7.2 Extreme Programming (XP).

Es una de las metodologías de desarrollo de software más exitosas en la actualidad, utilizadas para proyectos de corto plazo. La metodología consiste en una programación rápida o extrema, cuya particularidad es tener como parte del equipo, al usuario final, pues es uno de los requisitos para llegar al éxito del proyecto. (Sanchez, 2004)

Lo fundamental en este tipo de metodología es la comunicación, entre los usuarios y los desarrolladores, la simplicidad, al desarrollar y codificar los módulos del sistema, y la retroalimentación, concreta y frecuente del equipo de desarrollo, el cliente y los usuarios finales.

1.7.3 Microsoft Solution Framework (MSF).

Esta es una metodología flexible e interrelacionada con una serie de conceptos, modelos y prácticas de uso, que controlan la planificación, el desarrollo y la gestión de proyectos tecnológicos. MSF se centra en los modelos de proceso y de equipo dejando en un segundo plano las elecciones tecnológicas. (Sanchez, 2004)

MSF tiene como características que es flexible, adaptable, usado en cualquier parte. Además de escalable, puede organizar equipos, así como también, proyectos que requieren 50 personas a más.

1.7.4 Propuesta de la Metodología de desarrollo de software.

La Metodología de desarrollo de software que se utilizará para el proceso de la aplicación es la Metodología RUP, por ser más adaptable para proyectos de largo plazo, ya que la aplicación que se implementará pertenece al proyecto de la Intranet 2, el cual está desarrollándose en estos momentos, con el objetivo de renovar la Intranet actual que presenta muchos problemas en cuanto a arquitectura, diseño y organización de la información, proyecto que necesita tiempo, para crear una nueva arquitectura, seleccionar nuevas herramientas y tecnologías, y realizar un nuevo diseño. Por todas estas características no se utiliza la Metodología XP, que a pesar de ser una de las metodologías de desarrollo de software más exitosas en la actualidad se recomienda para proyectos de corto plazo.

Además, este proyecto debe seleccionar con cuidado las tecnologías que va utilizar en el mismo y viendo las ventajas que cada una presenta, para que el producto tenga la mejor calidad posible, seguridad y flexibilidad. Por lo que no se elige MSF, que se centra más bien en los modelos de proceso y de equipo dejando en un segundo plano las elecciones tecnológicas.

1.8 Conclusiones.

En este capítulo se han abordado los temas relacionados con el objeto de estudio, los procesos objeto de automatización, los objetivos de trabajo, las tendencias y tecnologías actuales, así como las herramientas a utilizar en el desarrollo de nuestra aplicación; con el objetivo de profundizar en el conocimiento de estos temas y fundamentar la necesidad de dar una solución informática a la situación existente en la Intranet de la Universidad de las Ciencias Informáticas para gestionar la publicación de los banners.

Se han expuesto las ventajas de las diferentes herramientas que se utilizarán en la implementación del sistema, y en la modelación de los diferentes diagramas.

Capítulo 2: Modelo del Negocio.

2.1 Introducción.

En este capítulo se detallarán los distintos pasos de la metodología propuesta en el capítulo anterior (RUP) para el desarrollo de la aplicación. Esto incluirá la elaboración del diagrama de casos de usos del negocio, definición de actores y trabajadores del negocio, las descripciones textuales de los casos de uso y la exposición de los requisitos de la plataforma.

Este es el primer flujo de trabajo que se realiza durante las fases de desarrollo de un sistema informático y tiene el objetivo de que se pueda comprender la estructura y la dinámica de la organización.

2.2 Actores y trabajadores del negocio.

Un actor del negocio es cualquier individuo, grupo, organización o máquina que interactúa con el negocio. El actor siempre permanece fuera de la frontera del negocio que se está investigando. En cambio, un trabajador del negocio representa a personas o sistemas (software) dentro del negocio que son los que realizan las actividades que están comprendidas dentro de un caso de uso.

Actores del Negocio

Actores	Descripción
Solicitante	Persona que realiza una solicitud para publicar un banner en la Intranet.

Tabla 2.1 – Actores del negocio.

Trabajadores del Negocio

Trabajadores	Descripción
Administración	Las personas que se encargan de aprobar o rechazar la solicitud (los compañeros que pertenecen a la dirección de informatización).
administrador	Persona encargada de guardar los datos de la solicitud para la publicación del banner.

Tabla 2.2 – Trabajadores del negocio.

2.3 Diagrama de Casos de Uso del Negocio.

Un caso de uso del negocio representa un proceso dentro del negocio que se estudia, por lo que se corresponde con una secuencia de acciones, con un orden lógico y que produce un resultado observable para ciertos actores del negocio.

En la siguiente figura se muestra el diagrama con los casos de uso del negocio que se han identificado en el sistema de gestión de banners.

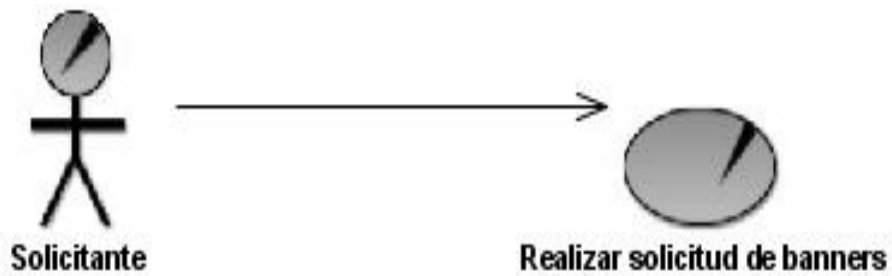


Figura 2.1 – Diagrama de casos de uso del negocio.

2.4 Descripción de los Casos de Uso del Negocio.

2.4.1 CUN Realizar Solicitud de Banners.

CASO DE USO DEL NEGOCIO		Realizar Solicitud de Banners.
Actores	Solicitante	
Propósito	Permitir que se pueda solicitar un banner para su publicación en la Intranet.	
<p>Resumen: El caso de uso comienza una vez que el solicitante le manda un correo a los compañeros que pertenecen a la dirección de informatización para que ellos le aprueben el banner, si ellos lo aprueban ya que cumple con los requisitos, el administrador de la Intranet es el encargado de publicarlo.</p>		
CURSO NORMAL DE LOS EVENTOS.		
ACCIÓN DEL ACTOR	RESPUESTA DEL PROCESO DE NEGOCIO	
1. El solicitante le envía la solicitud a la administración (a los compañeros de la dirección de informatización) para que le publiquen un banner con toda la información necesaria (dónde, cuándo, etc.)	2. La administración recibe la solicitud. 3. La administración analiza la solicitud. 4. La administración acepta la solicitud, el banner cumple con los requisitos establecidos. 5. La administración le envía la solicitud al administrador para la publicación del banner. 6. El administrador se encarga de guardar los datos de la solicitud para su publicación. Finaliza el Caso de Uso Realizar Solicitud de Banners.	
	CURSO ALTERNO DE LOS EVENTOS	
	4.1 La administración rechaza la solicitud, el banner no cumple con los requisitos establecidos. 5.1 Le notifica el rechazo de la solicitud al solicitante. Finaliza el Caso de Uso Realizar Solicitud de Banners.	
<i>Prioridad</i>	Crítico.	
<i>Mejoras</i>	-	

Tabla 2.3 – Descripción del CUN Realizar Solicitud de Banners.

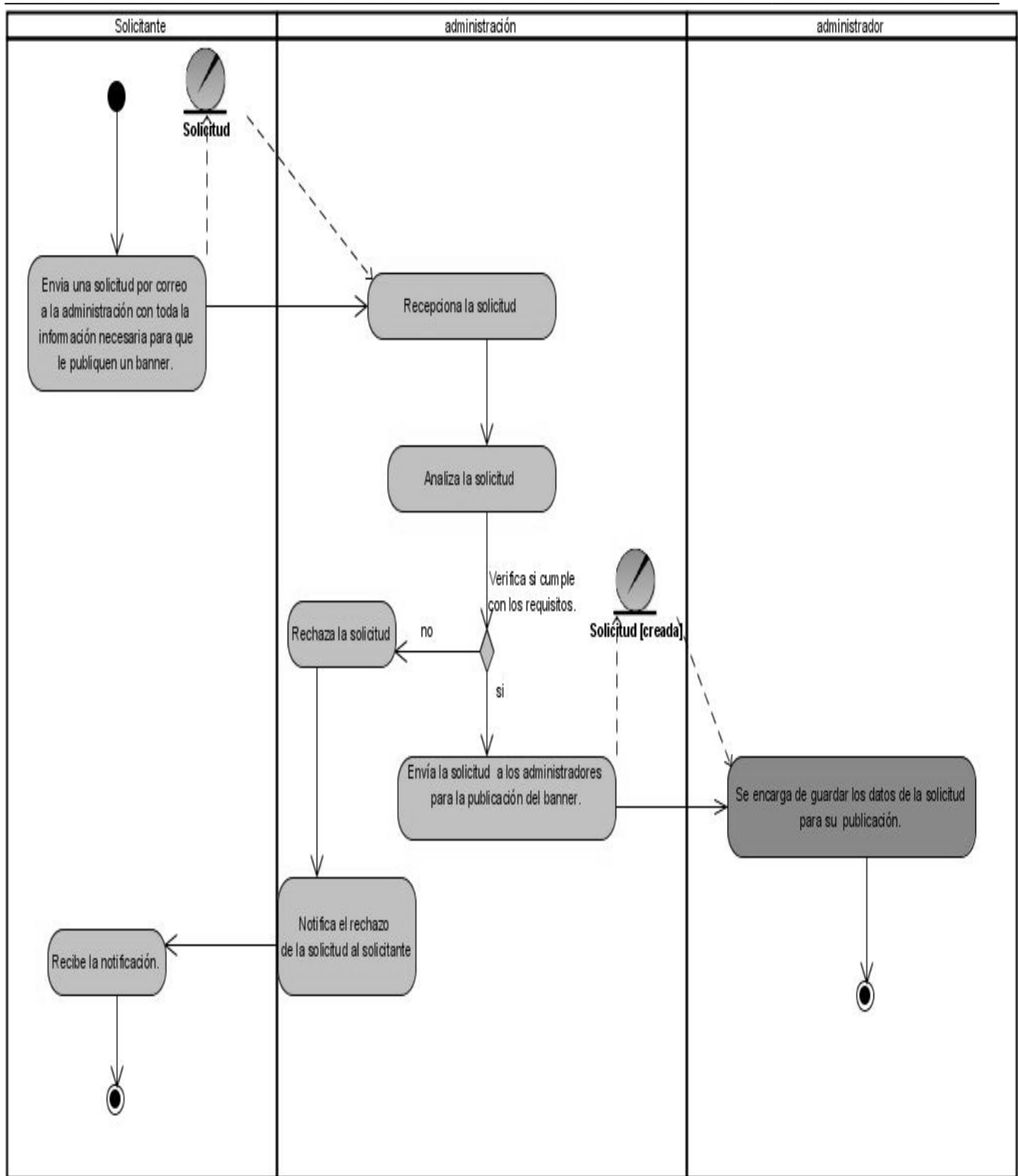


Figura 2.2 – Diagrama de actividad del CUN Realizar Solicitud de Banners.

2.5 Modelo de Objetos del Negocio.

El modelo de objeto es aquel diagrama que representa las entidades con que se relacionan los trabajadores del negocio. Una entidad del negocio no es más que algo físico que se utilice en el proceso de negocio y que sirva para obtener o actualizar información.

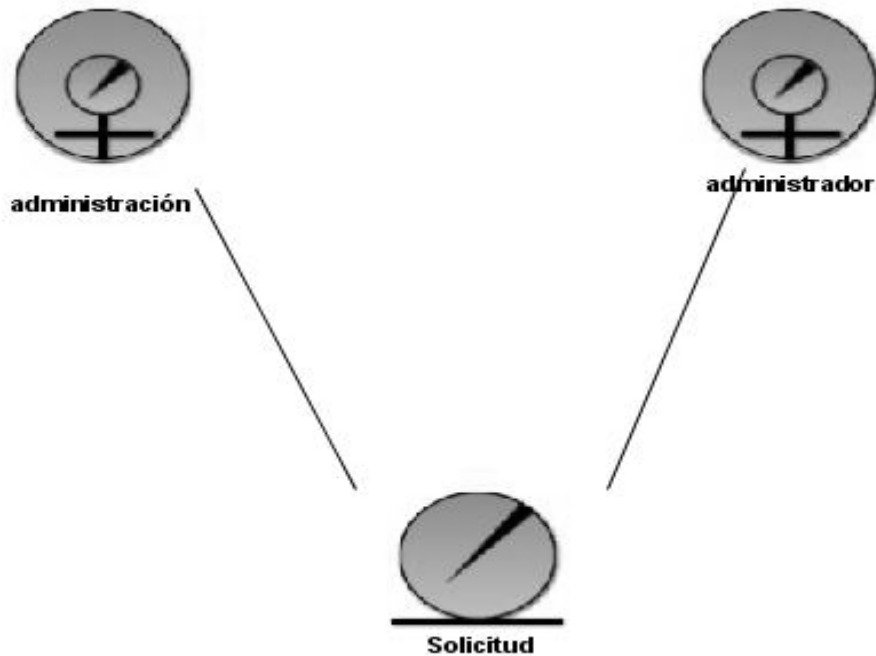


Figura 2.3 – Modelo de objetos del negocio.

2.6 Flujo de Requisitos.

Los requisitos son capacidades y condiciones con las cuales debe ser conforme el sistema. Pueden ser funcionales o no funcionales. Los requisitos del sistema se listan a continuación:

2.6.1 Requisitos Funcionales.

Los requisitos funcionales indican el comportamiento del sistema. Posteriormente estos requisitos son modelados a través del diagrama de casos de uso del sistema.

R1. Autenticar usuario.

- El sistema verifica que los datos entrados por el usuario sean los correctos para darle acceso a los servicios que brinda el sistema.
 - a) Nombre.
 - b) Contraseña.

R2. Insertar usuario al sistema.

- El sistema adiciona un nuevo usuario a la base de datos.
- a) Nombre
- b) Correo
- c) Privilegio
- d) Clave

R3. Eliminar usuario del sistema.

- El sistema elimina un usuario de la base de datos.

R4. Modificar datos del usuario.

- El sistema modifica los datos del usuario, actualizando la base de datos.
- a) Nombre.
- b) Correo.
- c) Privilegio.
- d) Clave.

R5. Insertar banners.

- El sistema adiciona un nuevo banner a la base de datos.
- a) Nombre.
- b) Estado (Activo, Inactivo).
- c) Categoría.
- d) Url del sitio.
- e) el sitio.

R6. Eliminar banners.

- El sistema elimina un banners de la base de datos.

R7. Modificar datos del banners.

- El sistema modifica los datos del banner, actualizando la base de datos.
- a) Categoría.
- b) Estado (Activo, Inactivo).
- c) Nombre.
- d) Url del sitio.

R8. Insertar nueva categoría de banners.

- El sistema adiciona una nueva categoría a la base de datos.

a) Nombre

R9. Mostrar catálogo de banners por categorías.

- El sistema muestra un listado de banners por categoría.

R10. Insertar planificación de banners.

- El sistema le agrega una planificación de publicación al banner.

a) Fecha de Publicación.

b) Fecha de Retirar el banner.

c) Frecuencia.

d) Hora de inicio.

e) Tipo de planificación (Diaria, Semanal, Mensual).

R11. Eliminar planificación de banners.

- El sistema elimina la planificación del banner de la base de datos.

R12. Modificar planificación de banners.

- El sistema modifica la planificación del banner de la base de datos.

a) Fecha de Publicación.

b) Fecha de Retirar el banner.

c) Frecuencia.

d) Hora de inicio.

e) Tipo de planificación (Diaria, Semanal, Mensual).

R13. Mostrar listado de banners con la fecha que se publicaran.

- El sistema muestra un listado de banners con las fechas de publicación.

R14. Mostrar listado de banners con la fecha que se retiraran del sitio.

- El sistema muestra un listado de banners con las fechas que se retiraran del sitio.

R15. Mostrar un listado de banners activos.

- El sistema muestra un listado de banners que su estado es activo.

R16. Mostrar un listado de no activos.

- El sistema muestra un listado de banners que su estado es no activo.

R17. Mostrar un listado de banners que aún no tienen fecha de publicación.

- El sistema muestra un listado de banners que aún no se le ha realizado ninguna planificación para su publicación.

2.6.2 Requisitos No Funcionales.

Los requerimientos no funcionales son propiedades o cualidades que el producto debe tener. Debe pensarse en estas propiedades como las características que hacen al producto atractivo, usable, rápido o confiable.

Usabilidad:

- Para utilizar el sistema es necesario poseer conocimientos elementales de computación y sobre el ambiente Web en sentido general.
- El sistema podrá ser usado por personas que estén autorizadas.

Apariencia o Interfaz Externa:

- El sistema deberá poseer una interfaz amigable y de fácil manejo, agradable y con colores adecuados.

Rendimiento:

- El tiempo de respuesta debe ser corto. El acceso a la información debe ser rápido y de la manera más fácil posible.

Portabilidad:

- El sistema debe ser capaz de instalarse en cualquier empresa o lugar.

Seguridad:

- El sistema debe identificar con certeza a los diversos usuarios que interactúan con él.
- El sistema tiene que garantizar que la información sea vista únicamente por quien tiene derecho a verla.

Soporte:

- El sistema contará con una ayuda para el usuario, con la cual podrá aprender rápidamente a utilizar la aplicación Web. Por otra parte, estará bien documentado para garantizar futuros mantenimientos.

2.7 Conclusiones.

En este capítulo se realizó la modelación del negocio, donde se obtuvo el diagrama de casos de uso del negocio, así como su descripción en formato expandido y el diagrama de actividad donde se sombrearon las actividades que se van a automatizar en esta primera versión del sistema. Se identificaron y describieron los actores y trabajadores del negocio, se obtuvo además el modelo de objetos del negocio y se identificaron los requisitos del sistema, obteniéndose así las funcionalidades que debe tener el mismo.

El desarrollo de este capítulo es de suma importancia debido a que con la modelación del negocio se asegura que los usuarios finales y desarrolladores tengan un entendimiento común de los procesos que se llevan a cabo y es en esta primera fase de desarrollo donde se determina la funcionalidad que va a tener el sistema.

Capítulo 3: Características del Sistema.

3.1 Introducción.

En este tercer Capítulo se hace una descripción del sistema y se proponen los rasgos generales de solución futura. Se exponen aquí los actores del sistema así como la interacción de cada uno de ellos con el sistema. Se presenta el modelo y el diagrama de casos de uso del sistema y la descripción de cada uno de estos casos de uso en modelo expandido.

3.2 Actores del sistema a automatizar.

Los actores del sistema representan entidades externas que interactúan directamente con el sistema (humanos, máquinas u otros sistemas).

Nombre del Actor	Justificación
Administrador	Administrador del sistema realiza la gestión de los banners, permite el control y acceso al sistema.

Tabla 3.1 – Actores del sistema.

3.3 Casos de Usos del Sistema.

Los casos de uso del sistema que aparecen a continuación tienen como objetivo satisfacer los requisitos funcionales descritos con anterioridad.

CU - 1	Autenticar Usuario
Actor	Administrador
Descripción	El usuario se identifica ante el sistema para recibir los permisos que hayan sido definidos para él.
Referencia	R1

Tabla 3.2 – Caso de Uso Autenticar Usuario.

CU - 2	Gestionar Usuario
Actor	Administrador
Descripción	El administrador accede al sistema para insertar, modificar o eliminar un usuario
Referencia	R2, R3, R4

Tabla 3.3 – Caso de Uso Gestionar Usuario.

CU - 3	Gestionar Banners
Actor	Administrador
Descripción	El administrador accede al sistema para insertar, modificar, eliminar o mostrar el catálogo de banners por categoría.
Referencia	R5, R6, R7. R8, R9

Tabla 3.4 – Caso de Uso Gestionar Banners.

CU - 4	Gestionar Planificación
Actor	Administrador
Descripción	El administrador accede al sistema para insertar, modificar o eliminar una planificación.
Referencia	R10, R11, R12

Tabla 3.5 – Caso de Uso Gestionar Planificación.

CU - 5	Gestionar Reportes
Actor	Administrador
Descripción	El administrador accede al sistema para mostrar unos de los reportes.
Referencia	R13, R14, R15, R16, R17

Tabla 3.6 – Caso de Uso Gestionar Reportes.

3.4 Diagrama de Casos de Uso del Sistema.

Un caso de uso del sistema describe la secuencia posible de interacciones entre el sistema y uno o más actores como consecuencia de un estímulo inicial de alguno de los actores.

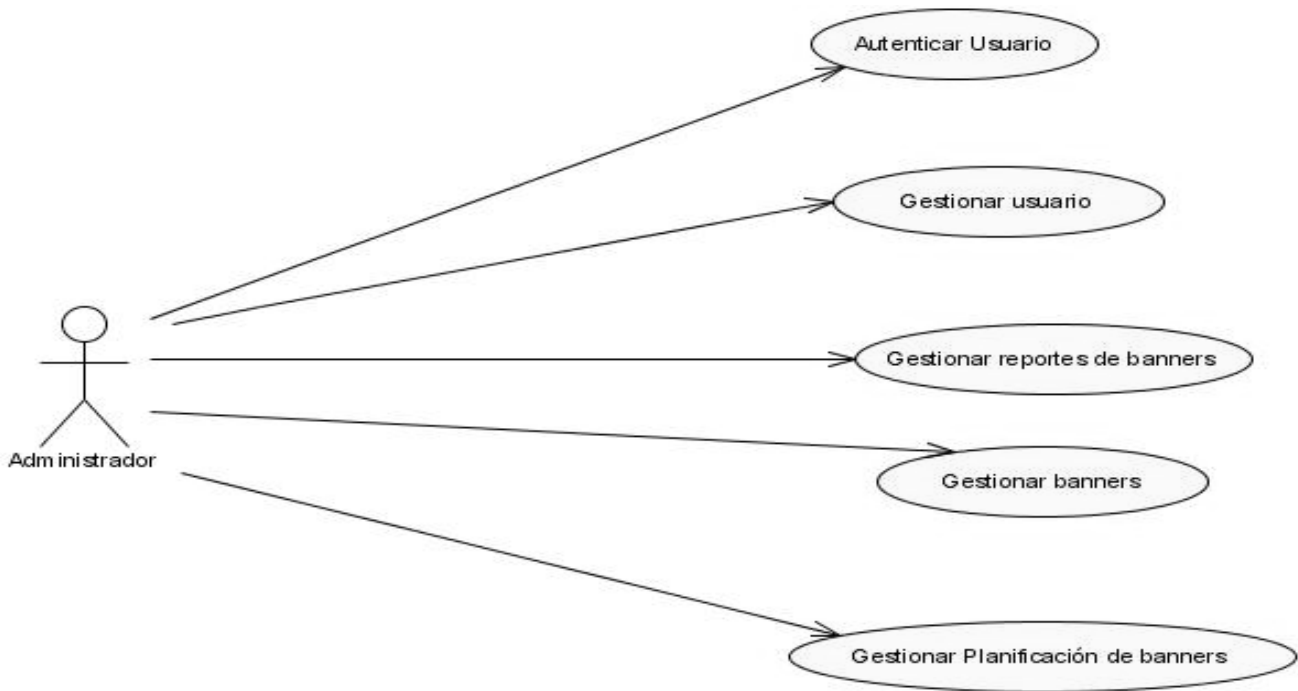


Figura 3.1 – Diagrama de casos de uso del sistema.

3.4 Descripción de los Casos de Uso del Sistema.

Las descripciones de los casos de uso del sistema permiten detallar el flujo de acciones que se produce con la interacción de los actores y los casos de uso, haciendo referencia a aquellas transacciones similares, requisitos, que conforman los casos de uso del sistema.

CU-1	Autenticar Usuario
Actores:	Administrador
Propósito:	Acceder al sistema.

Capítulo 3: Características del Sistema.

<u>Resumen:</u> El caso de uso se inicia cuando el usuario introduce el nombre de usuario y su contraseña para acceder a los servicios que le brinda el sistema, el sistema verifica que el usuario existe.	
Referencias:	R1
Precondiciones:	-
CURSO NORMAL DE LOS EVENTOS	
<u>Acción del actor</u>	<u>Respuesta del sistema</u>
1) El usuario entra su nombre de usuario y su contraseña.	2) El sistema verifica que los datos insertados sean los correctos.
	3) El sistema da acceso al usuario a los servicios. Finaliza el Caso de Uso Autenticar Usuario.
CURSO ALTERNO DE LOS EVENTOS 1	
Viene del paso 3 del curso normal de los eventos.	
	1) Los datos están incorrectos. Se retorna al paso 1 del curso normal de los eventos.
Poscondiciones:	-

Tabla 3.7 – CUS Autenticar Usuario.

CU-2	Gestionar Usuario
Actores:	Administrador
Propósito:	Controlar el acceso a los servicios que brinda el sistema.
<u>Resumen:</u> El caso de uso se inicia cuando el usuario desea eliminar, modificar o insertar un usuario al sistema.	
Referencias:	R2, R3, R4
Precondiciones:	Usuario del sistema autenticado.
CURSO NORMAL DE LOS EVENTOS	

<u>Acción del actor</u>	<u>Respuesta del sistema</u>
Sección Insertar usuario	
1) El usuario selecciona la opción Insertar usuario.	2) El sistema muestra un formulario para que el usuario introduzca los datos del usuario que desea adicionar al sistema.
3) El usuario entra los datos(nombre, correo, clave, privilegio)	4) El sistema guarda los datos entrados por el usuario y actualiza la base de datos. Finaliza la Sección Insertar usuario.
Sección Eliminar usuario	
1) El usuario selecciona la opción Eliminar usuario.	2) El sistema muestra el listado de los usuarios que están registrados en el sistema.
3) El usuario escoge el o los que desea eliminar y da la opción eliminar.	4) El sistema verifica que se desea eliminar el usuario.
5) El usuario confirma la eliminación.	6) El sistema lo elimina de la base de datos. Finaliza la Sección Eliminar usuario.
CURSO ALTERNO DE LOS EVENTOS 1	
Viene del paso 5 del curso normal de los eventos.	

1) El usuario no confirma la eliminación.	2) El sistema no realiza ninguna acción. Se retorna al paso 2 del curso normal de los eventos.
Sección Modificar datos del usuario	
1) El usuario escoge la opción Modificar sus datos.	2) El sistema muestra un listado con todos los usuarios que están insertados en el sistema.
3) El usuario escoge el que desea modificar sus datos.	4) El sistema muestra el formulario con las características del usuario escogido.
5) El usuario modifica los datos (nombre, correo, privilegio, clave), y envía el formulario al sistema.	6) El sistema modifica las características del usuario en la base de datos. Finaliza la Sección Modificar datos del usuario.
Poscondiciones:	-

Tabla 3.8 – CUS Gestionar Usuario.

CU-3	Gestionar Banners
Actores:	Administrador
Propósito:	Gestionar la manipulación de los banners en el sistema.
<u>Resumen:</u> El caso de uso comienza cuando el usuario selecciona la opción gestionar banners, y escoge las opciones que le aparecen, en las cuales podrá insertar banner, eliminar banner, insertar una nueva categoría de banner, modificar los datos del banner y mostrar el catálogo de banner por categoría.	

Referencias:	R5, R6, R7. R8, R9
Precondiciones:	Usuario del sistema autenticado.
CURSO NORMAL DE LOS EVENTOS	
<u>Acción del actor</u>	<u>Respuesta del sistema</u>
Sección Insertar Banner	
1) El usuario escoge la opción Insertar Banner.	2) El sistema muestra un formulario que el usuario debe llenar para guardar los datos del nuevo banner.
3) El usuario entra los datos del banner (el nombre del banner, el estado, el url del sitio y la categoría). Y le envía ese formulario al sistema.	4) El sistema guarda los datos del banner introducidos por el usuario. Adicionando un nuevo banner a la lista de banners. Finaliza la Sección Insertar Banner.
CURSO ALTERNO DE LOS EVENTOS 1	
Viene del paso 3 del curso normal de los eventos.	
1) El usuario no desea seleccionar una de las categorías que aparecen en el formulario, y entra un nuevo nombre. Se retorna al paso 4 del curso normal de los eventos.	

Sección Eliminar banner	
1) El usuario escoge la opción Eliminar banner.	2) El sistema muestra un listado con todos los banners.
3) El usuario escoge el banner que desea eliminar y da la opción eliminar.	4) El sistema verifica que se desea eliminar el banner.
5) El usuario confirma la eliminación.	6) El sistema lo elimina de la base de datos. Finaliza la Sección Eliminar Banner.
CURSO ALTERNO DE LOS EVENTOS 1	
Viene del paso 5 del curso normal de los eventos.	
1) El usuario no confirma la eliminación	2) El sistema no realiza ninguna acción. Se retorna al paso 2 del curso normal de los eventos.
Sección Modificar datos del banner	
1) El usuario escoge la opción Modificar datos del banner.	2) El sistema muestra un listado con todos los banners.
3) El usuario escoge el banner que desea modificar sus datos.	4) El sistema muestra el formulario con las características del banner que el usuario escogió.

Capítulo 3: Características del Sistema.

5) El usuario modifica los datos (categoría, estado, nombre, url del sitio), y envía el formulario al sistema.	6) El sistema modifica las características del banner en la base de datos. Finaliza la Sección Modificar datos del banner.
Sección Mostrar Catálogo de banner por categoría.	
1) El usuario escoge la opción Mostrar Catálogo de banner por categoría.	2) El sistema muestra las categorías de banners que tiene el sistema con el listado de banners que tiene cada uno. Finaliza la Sección Mostrar Catálogo de banner por categoría.
Poscondiciones:	-

Tabla 3.9 – CUS Gestionar Banners.

CU-4	Gestionar Planificación de banners.	
Actores:	Administrador	
Propósito:	Controlar la salida de los banners.	
<u>Resumen:</u> El caso de uso comienza cuando el usuario selecciona la opción Gestionar Planificación, en la cual podrá para asignarle una planificación al banner, eliminarla o modificarla.		
Referencias:	R10, R11, R12	
Precondiciones:	Usuario del sistema autenticado.	
CURSO NORMAL DE LOS EVENTOS		
<u>Acción del actor</u>	<u>Respuesta del sistema</u>	
Sección Insertar planificación		

1) El usuario selecciona la opción Insertar planificación.	2) El sistema muestra un listado con todos los banners.
3) El usuario escoge el banner que desea añadirle una planificación.	4) El sistema muestra un formulario donde el usuario debe introducir los datos de la planificación.
5) El usuario introduce los datos en el formulario para la salida del banner (fecha inicio, fecha fin, hora de inicio, tipo de planificación y frecuencia diaria), y se lo envía al sistema.	6) El sistema guarda los datos de la planificación en la base de datos. Finaliza la Sección Insertar planificación.
Sección Eliminar planificación	
1) El usuario selecciona la opción Eliminar planificación.	2) El sistema muestra un listado de todos banners con sus planificaciones.
3) El usuario escoge la planificación del banner que desea eliminar.	4) El sistema solicita al usuario la confirmación de la eliminación de la planificación.
5) El usuario confirma la eliminación.	6) El sistema elimina esa planificación para el banner. Finaliza la Sección Eliminar planificación.
CURSO ALTERNO DE LOS EVENTOS 1	
Viene del paso 5 del curso normal de los eventos.	
1) El usuario no confirma la eliminación.	2) El sistema no realiza ninguna acción. Se retorna al paso 2 del curso normal de los

	eventos.
Sección Modificar planificación	
1) El usuario selecciona la opción Modificar planificación.	2) El sistema muestra un listado de todos banners con sus planificaciones
3) El usuario escoge la planificación del banner que desea modificar.	4) El sistema le muestra al usuario un formulario con los datos de la planificación del banner que el escogió.
5) El usuario modifica los datos (fecha inicio, fecha fin, hora de inicio, tipo de planificación y frecuencia diaria), y se lo envía al sistema.	6) El sistema modifica los datos de la planificación. Finaliza la Sección Modificar planificación.
Poscondiciones:	–

Tabla 3.10 – CUS Gestionar Planificación de Banners.

CU-5	Gestionar Reportes de banners.	
Actores:	Administrador	
Propósito:	Mostrar listados del estado y planificación de los banners.	
<u>Resumen:</u> El caso de uso se inicia cuando el usuario escoge la opción Gestionar reportes de banners, para que el sistema muestre diferentes reportes.		
Referencias:	R13, R14, R15, R16, R17	
Precondiciones:	Usuario del sistema autenticado.	
CURSO NORMAL DE LOS EVENTOS		
<u>Acción del actor</u>	<u>Respuesta del sistema</u>	
1) El usuario desea mostrar uno de los	2) El sistema le muestra al usuario el reporte que el seleccionó. Finaliza el Caso de Uso Gestionar	

Capítulo 3: Características del Sistema.

siguientes reportes: a) Mostrar un listado de banners con la fecha que se publicaran. b) Mostrar un listado de banners con la fecha que se retiraran del sitio. c) Mostrar un listado de banners activos. d) Mostrar un listado de banners inactivos. e) Mostrar un listado de banners que aún no tienen fecha de publicación.	reportes de banners.
Poscondiciones:	-

Tabla 3.11 – CUS Gestionar Reportes de Banners.

3.6 Conclusiones.

En este Capítulo se mostró el diagrama de casos de uso del sistema, describiendo textualmente cada uno de ellos, así como la justificación de los actores del sistema que se relacionaran, dando de esta forma una vista global de cómo está concebido y cómo va a funcionar el sistema.

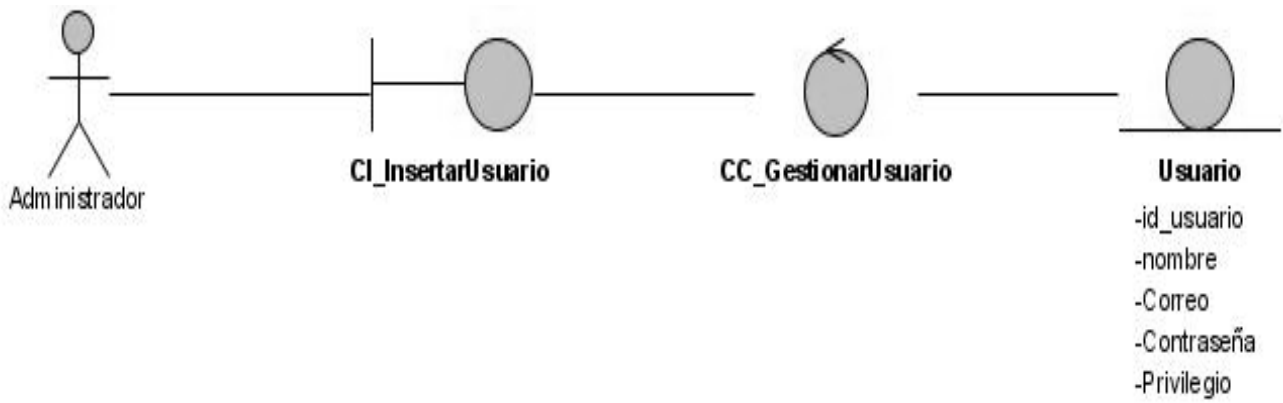


Figura 4.2 – Diagrama de clases de Análisis del CU Gestionar Usuario. Sección Insertar Usuario.

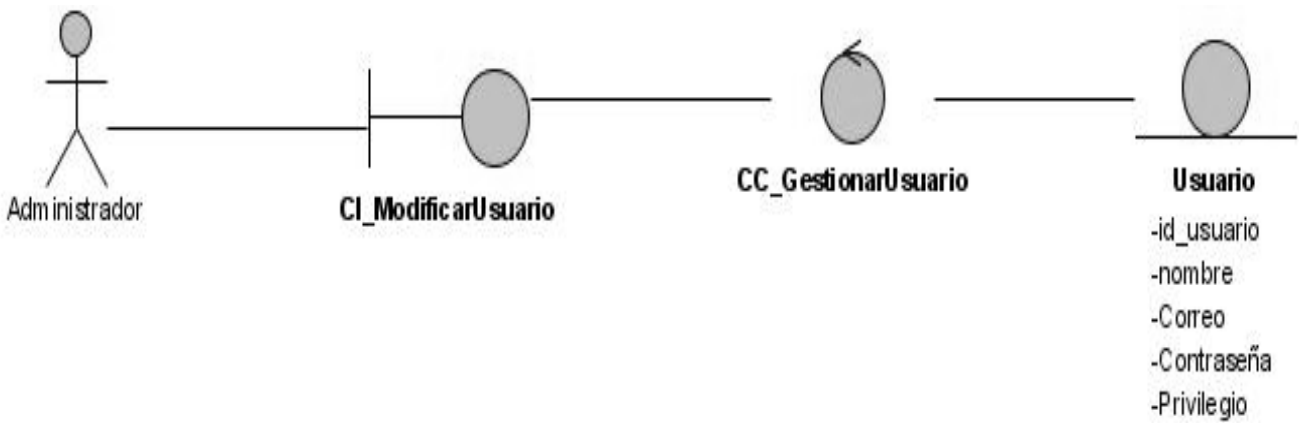


Figura 4.3 – Diagrama de clases de Análisis del CU Gestionar Usuario. Sección Modificar Usuario.

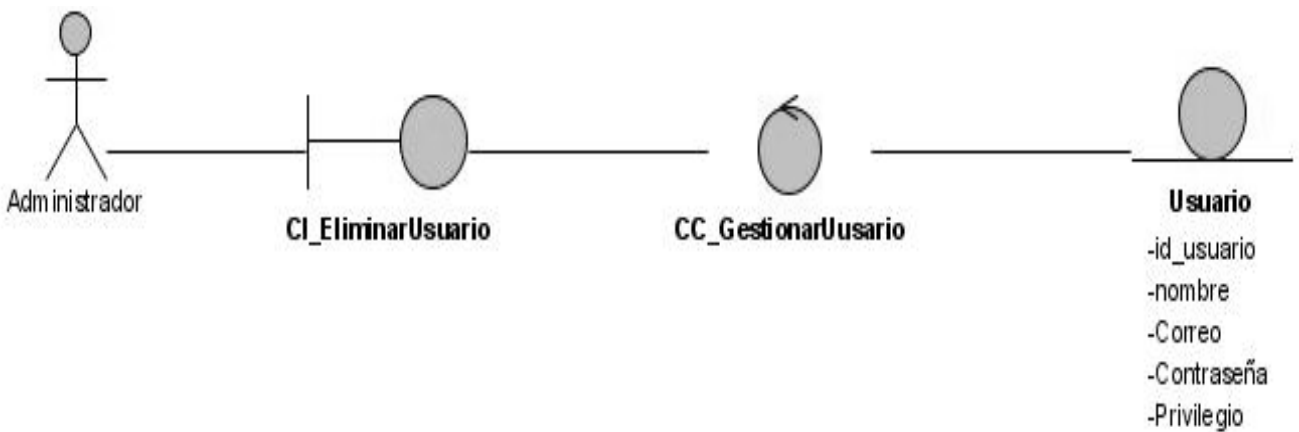


Figura 4.4 – Diagrama de clases de Análisis del CU Gestionar Usuario. Sección Eliminar Usuario.

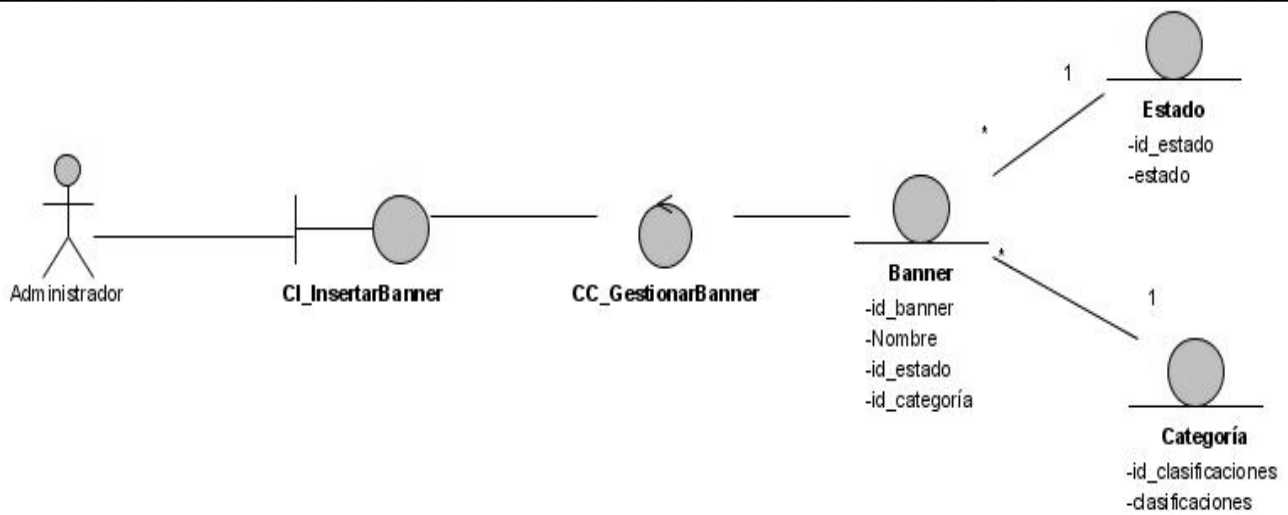


Figura 4.5 – Diagrama de clases de Análisis del CU Gestionar Banners. Sección Insertar Banner.

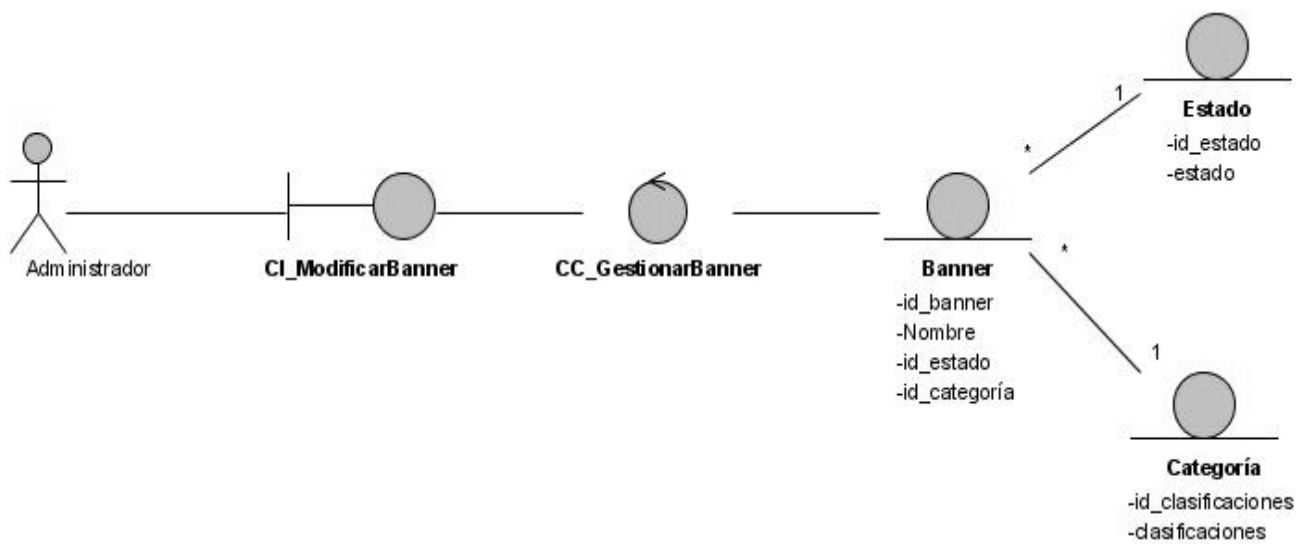


Figura 4.6 – Diagrama de clases de Análisis del CU Gestionar Banners. Sección Modificar Banners.

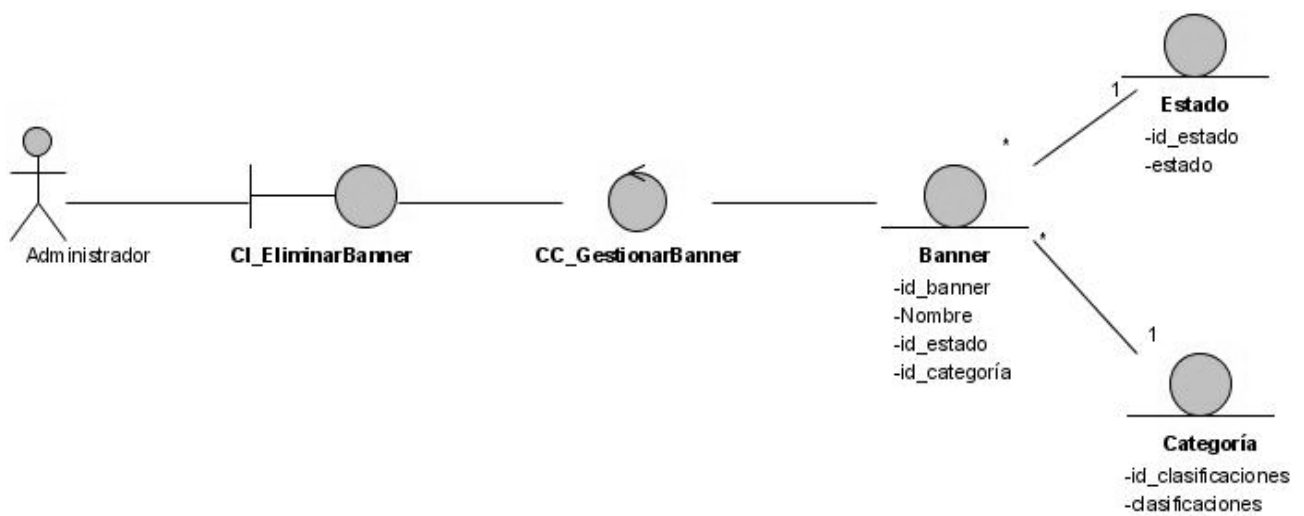


Figura 4.7 – Diagrama de clases de Análisis del CU Gestionar Banners. Sección Eliminar Banner.

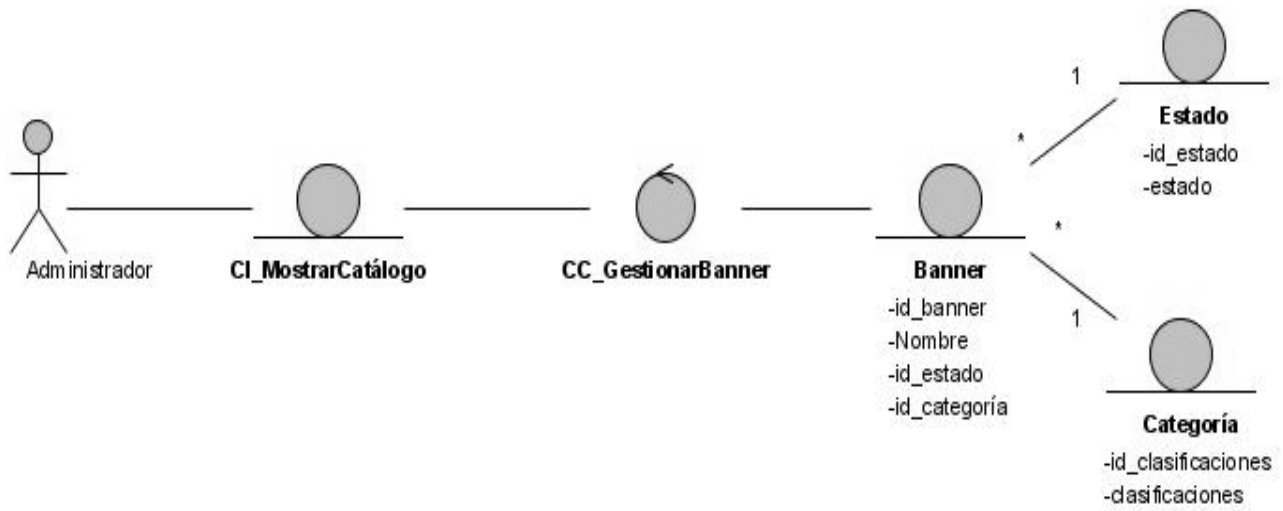


Figura 4.8 – Diagrama de clases de Análisis del CU Gestionar Banners. Sección Mostrar Catálogo de Banner por Categoría.

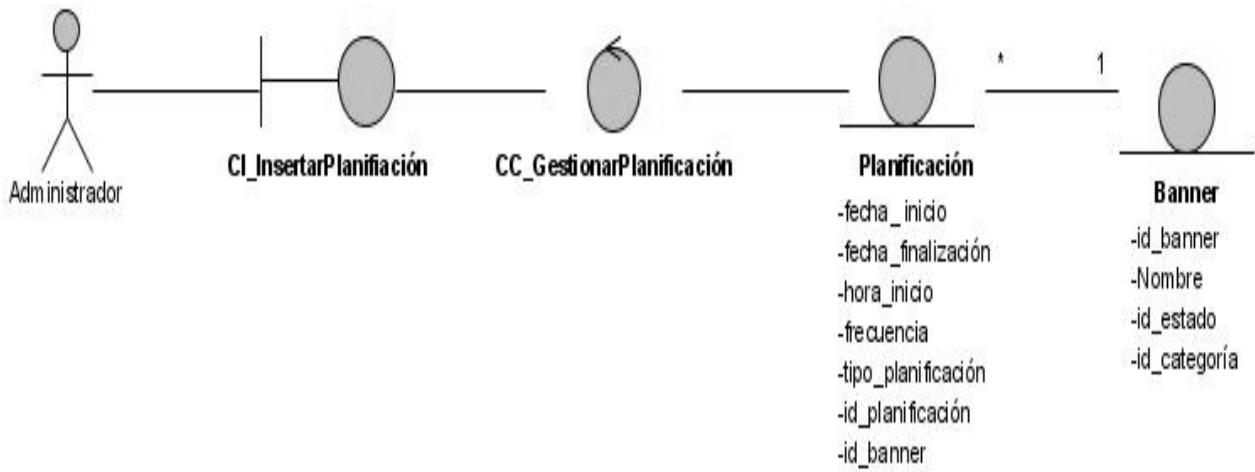


Figura 4.9 – Diagrama de clases de Análisis del CU Gestionar Planificación. Sección Insertar Planificación.

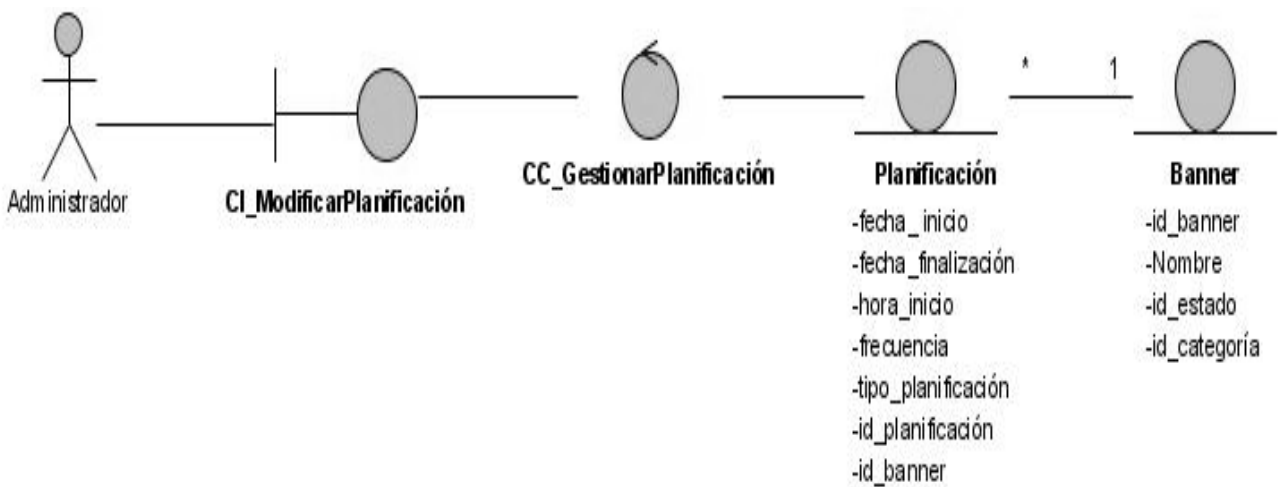


Figura 4.10 – Diagrama de clases de Análisis del CU Gestionar Planificación. Sección Modificar Planificación.

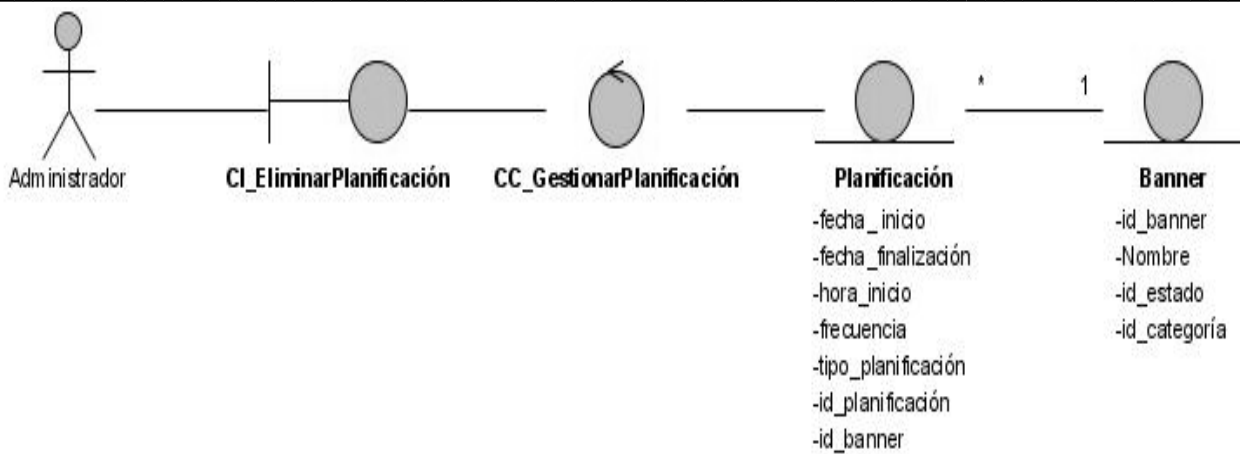


Figura 4.11 – Diagrama de clases de Análisis del CU Gestionar Planificación. Sección Eliminar Planificación.

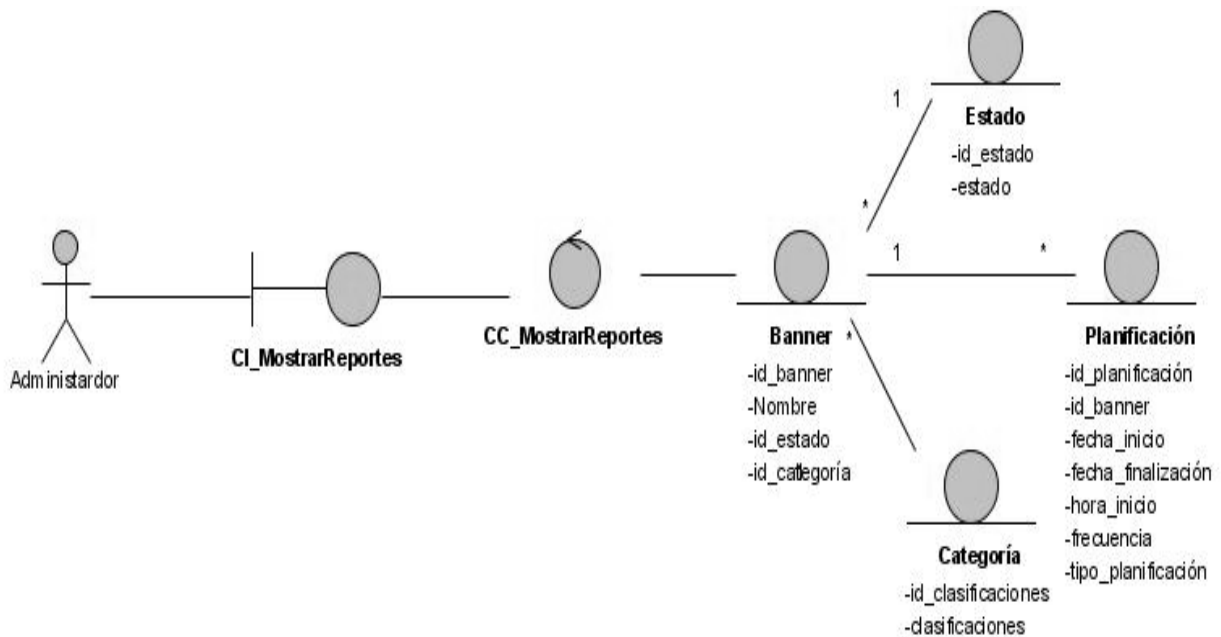


Figura 4.12 – Diagrama de clases de Análisis del CU Gestionar Reportes.

4.2.2 Diagramas de Interacción.

El lenguaje utilizado para ilustrar los diseños es, principalmente, los diagramas de interacción. UML incluye los diagramas de interacción para ilustrar el modo en el que los objetos interaccionan por medio de mensajes.

EL término diagrama de interacción es una generalización de dos tipos de diagramas UML más especializados; ambos pueden utilizarse para representar de forma similar interacciones de mensajes: Diagramas de colaboración y diagramas de secuencia.

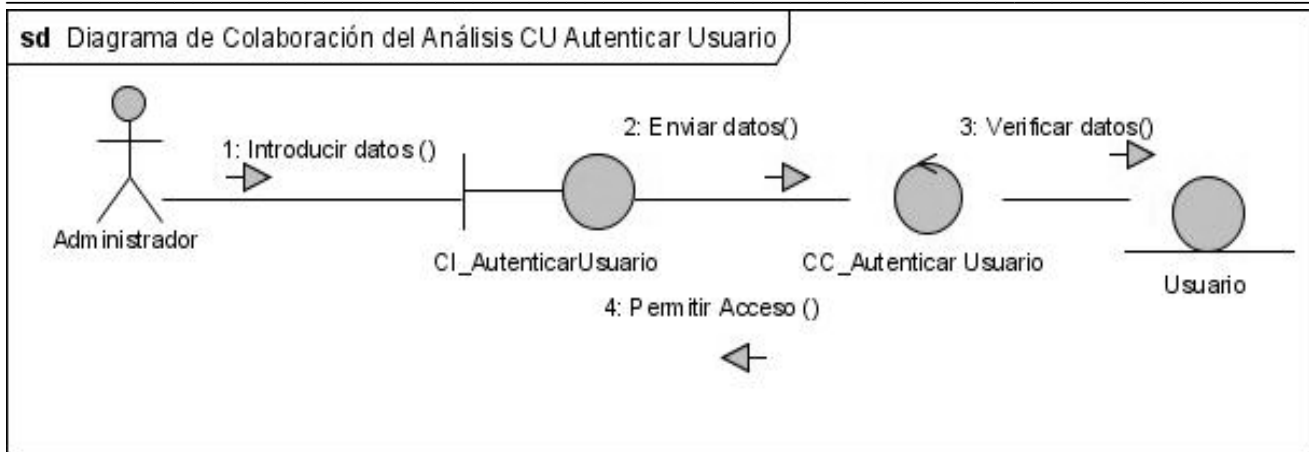


Figura 4.13 – Diagrama de Colaboración CU Autenticar Usuario.

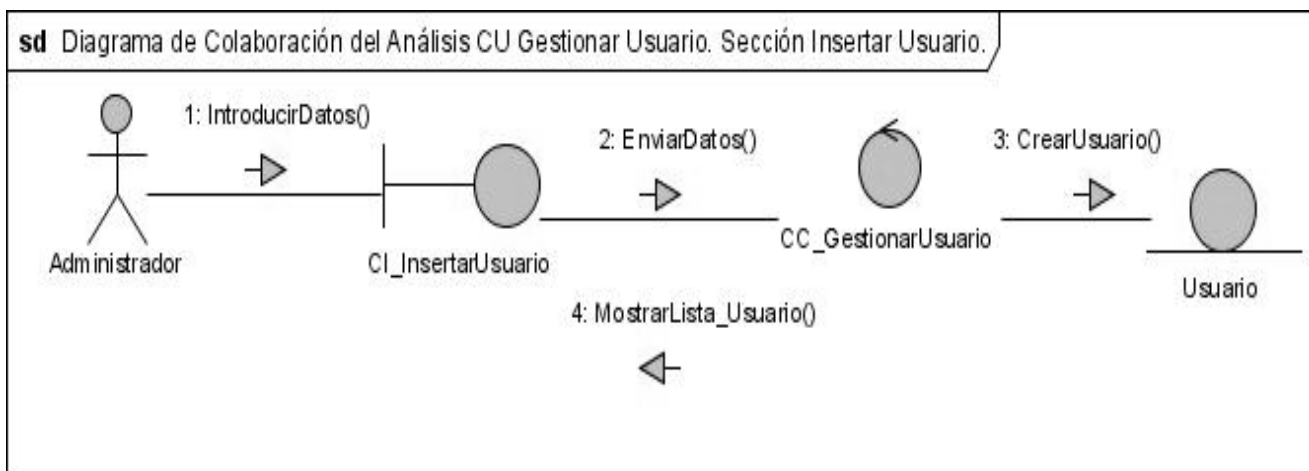


Figura 4.14 – Diagrama de Colaboración CU Gestionar Usuario. Sección Insertar Usuario.

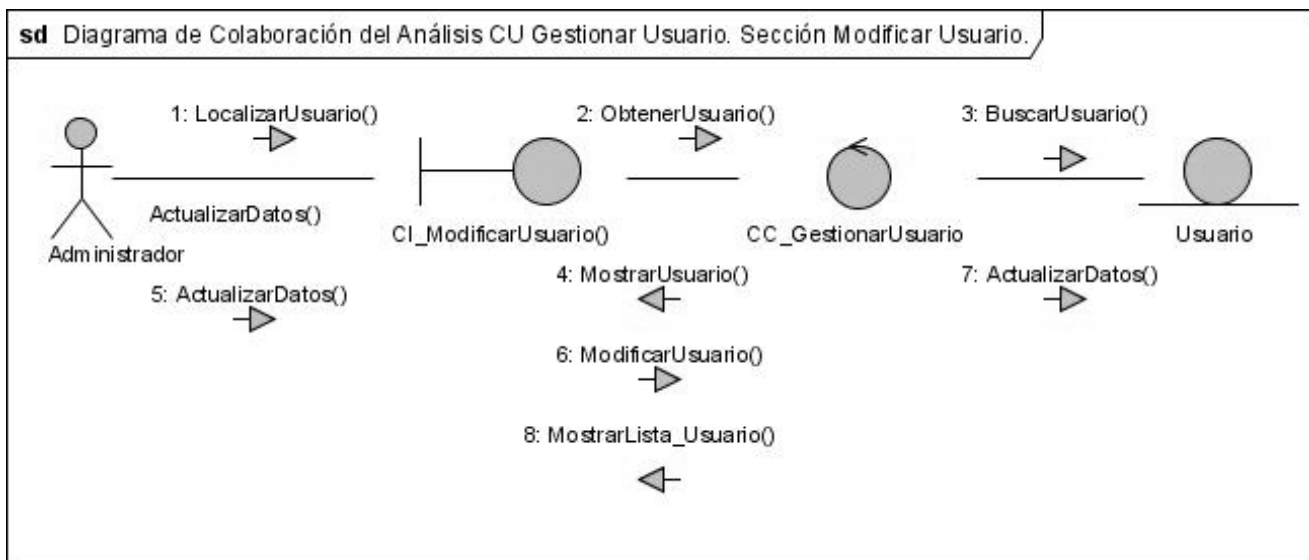


Figura 4.15 – Diagrama de Colaboración CU Gestionar Usuario. Sección Modificar Usuario.

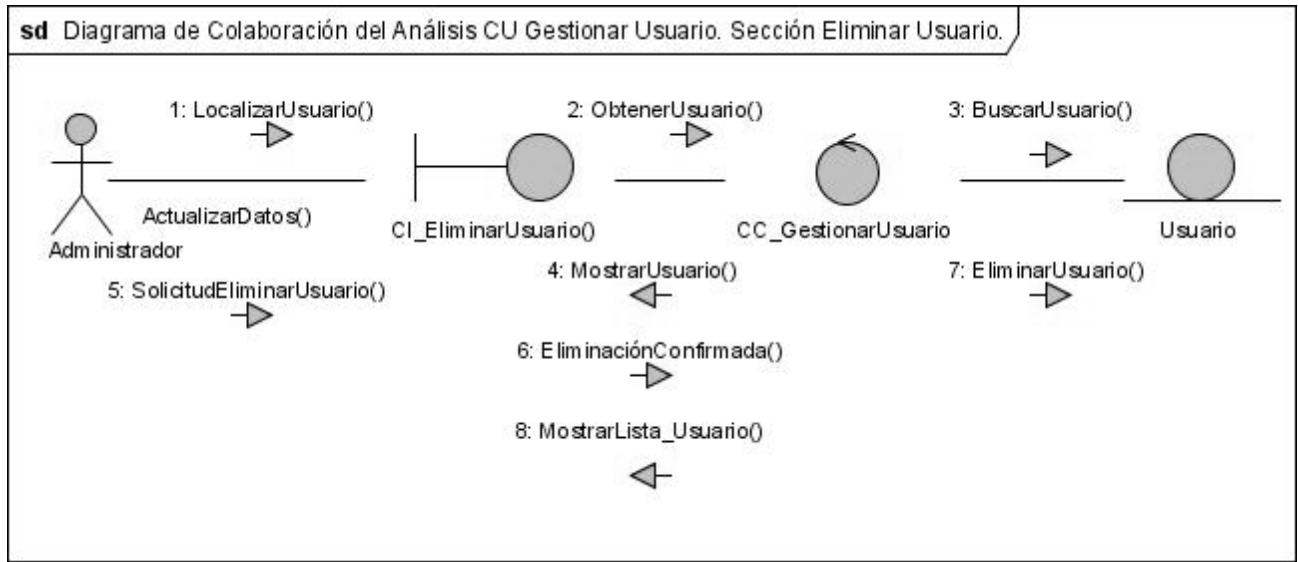


Figura 4.16 – Diagrama de Colaboración CU Gestionar Usuario. Sección Eliminar Usuario.

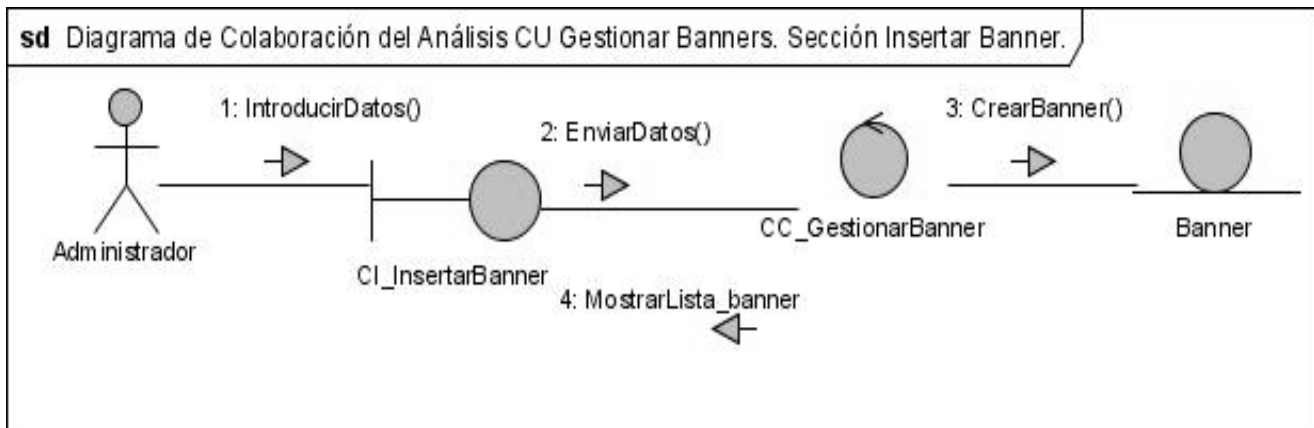


Figura 4.17 – Diagrama de Colaboración CU Gestionar Banners. Sección Insertar Banner.

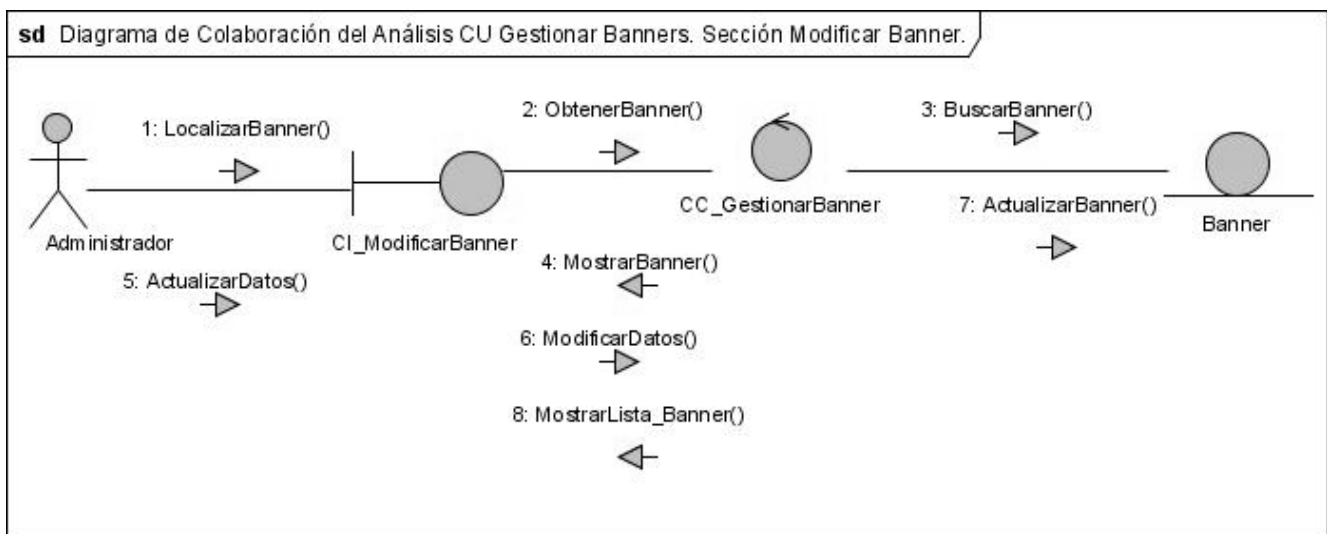


Figura 4.18 – Diagrama de Colaboración CU Gestionar Banners. Sección Modificar Banner.

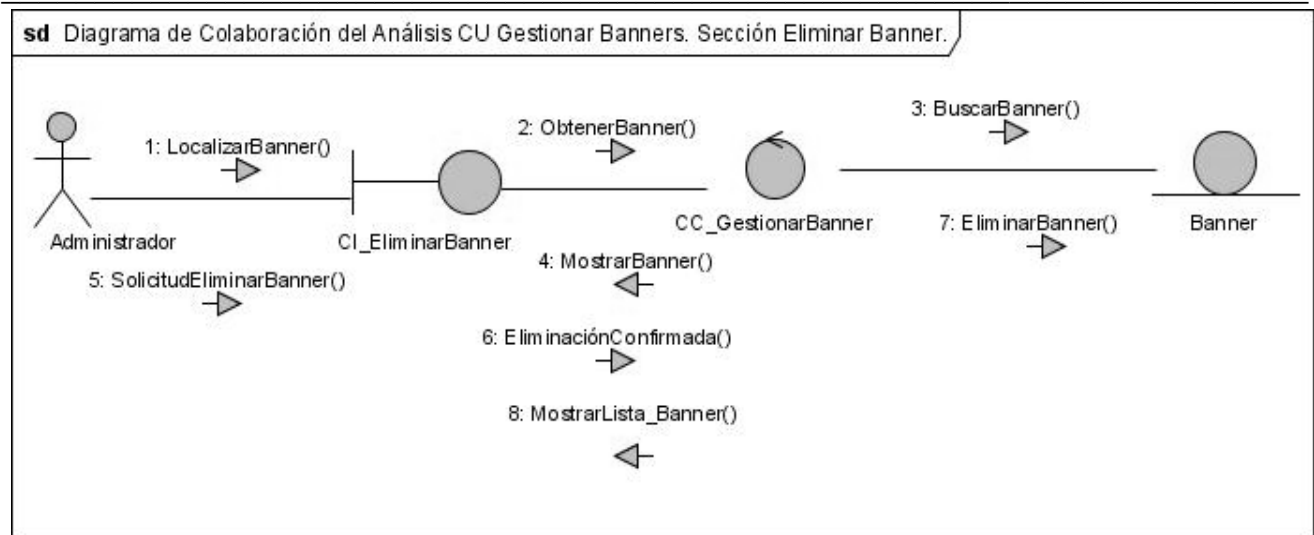


Figura 4.19 – Diagrama de Colaboración CU Gestionar Banners. Sección Eliminar Banner.

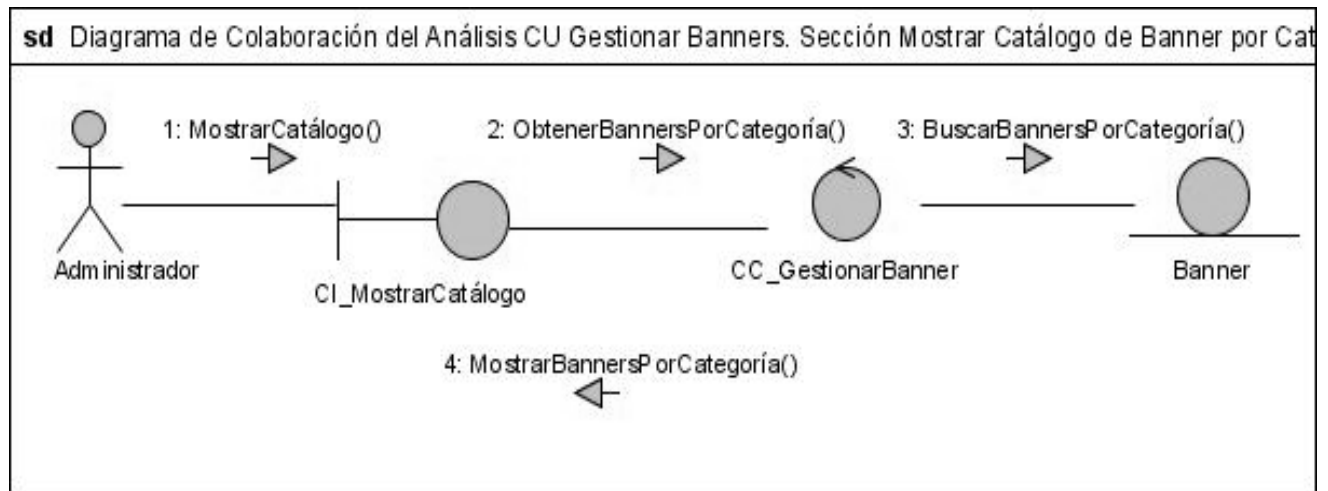


Figura 4.20 – Diagrama de Colaboración CU Gestionar Banners. Sección Mostrar Catálogo de Banner por Categoría.

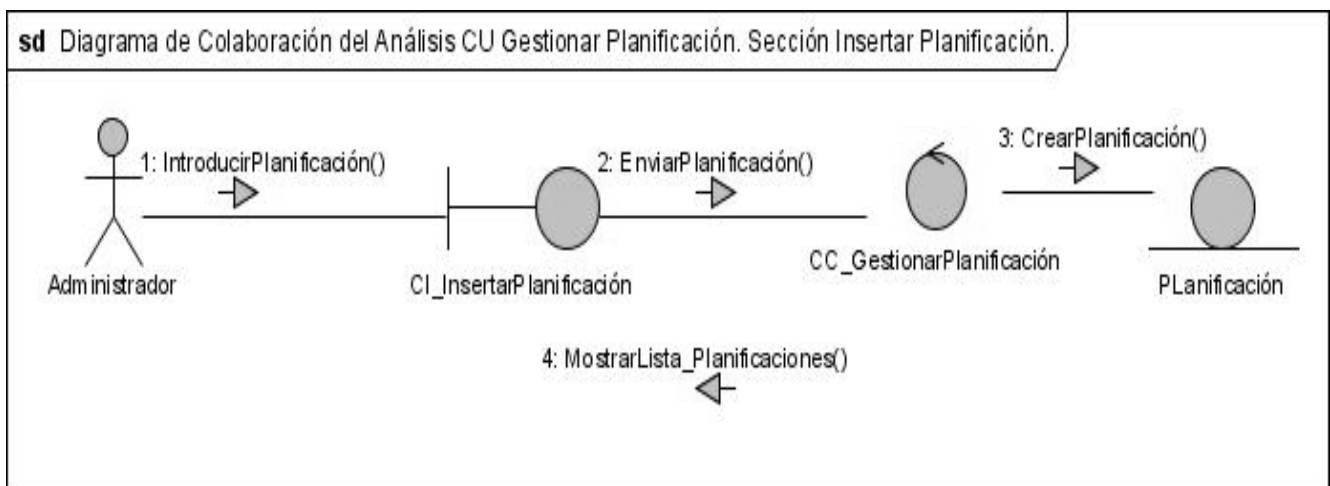


Figura 4.21 – Diagrama de Colaboración CU Gestionar Planificación. Sección Insertar Planificación.

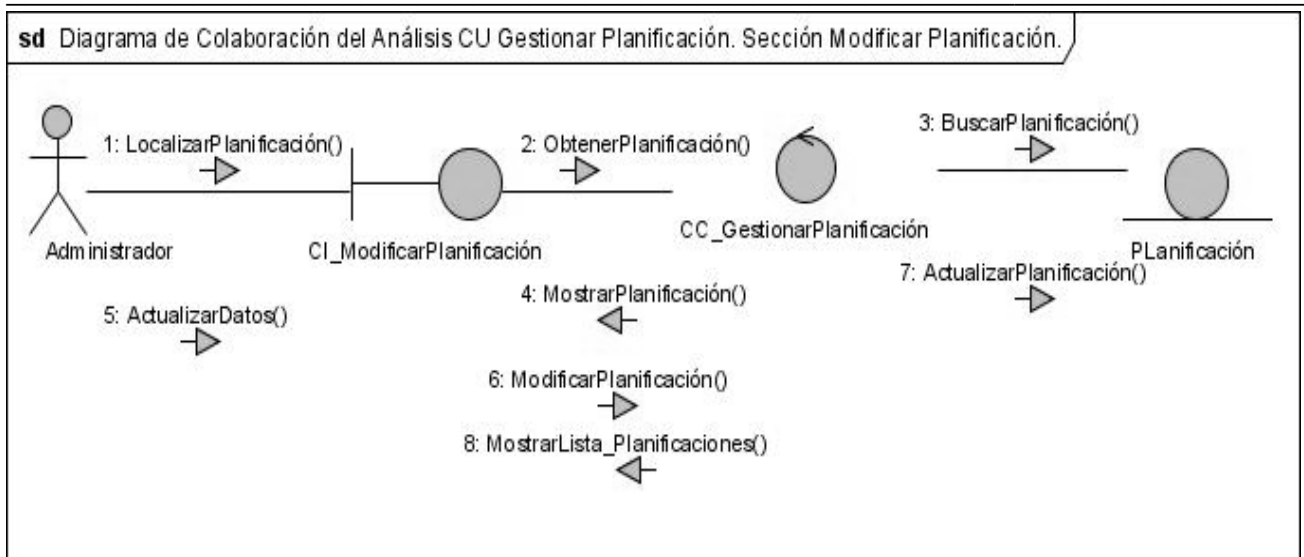


Figura 4.22 – Diagrama de Colaboración CU Gestionar Planificación. Sección Modificar Planificación.

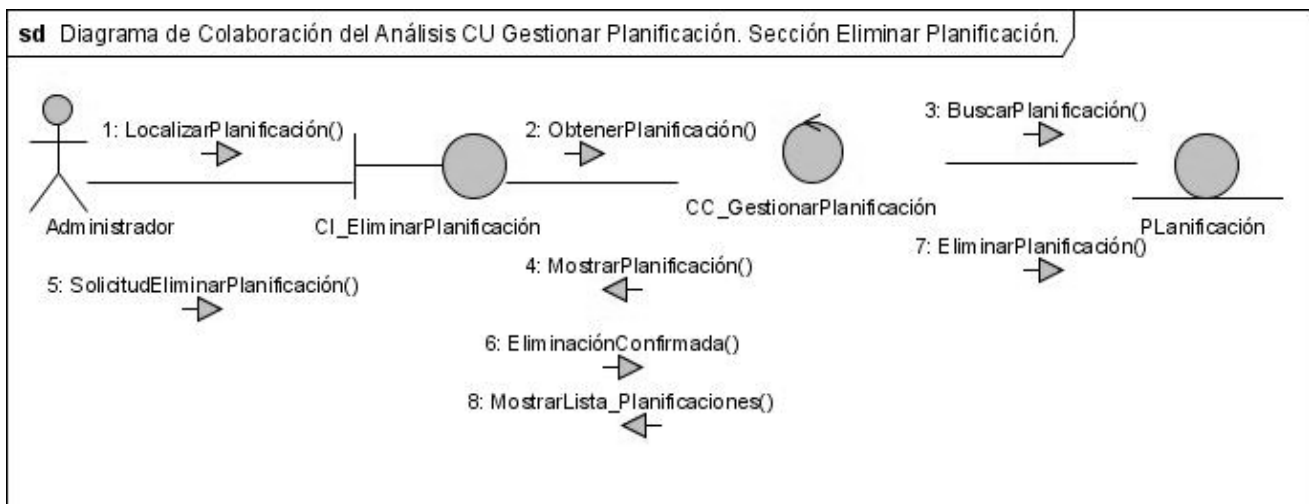


Figura 4.23 – Diagrama de Colaboración CU Gestionar Planificación. Sección Eliminar Planificación.

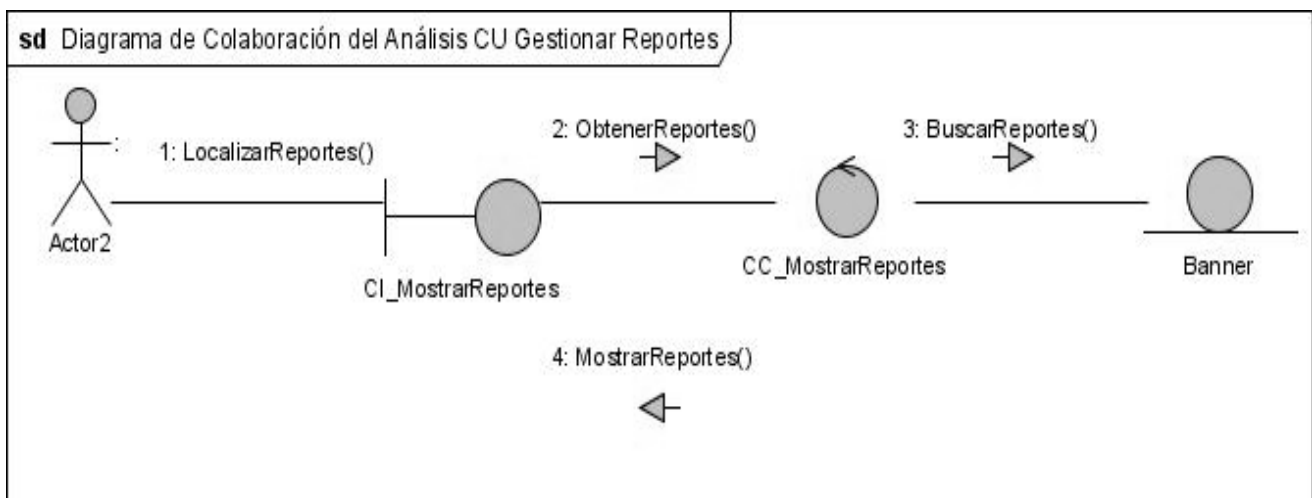


Figura 4.24 – Diagrama de Colaboración CU Gestionar Reportes.

4.3 Diagramas de Clases del Diseño Web.

El diagrama de diseño Web describe la realización física de los casos de uso, centrándose en como los requisitos funcionales y no funcionales junto con otras restricciones relacionadas con el entorno de implementación, tienen impacto en el sistema. Se muestran los atributos y métodos de cada clase, y se representa de una forma sencilla como se lleva a cabo la colaboración y las responsabilidades de las distintas clases que forman el sistema. Se modelan las páginas, los enlaces entre estas, todo el código que irá creando las páginas, así como el contenido dinámico de las mismas, una vez que estén en el navegador del cliente.

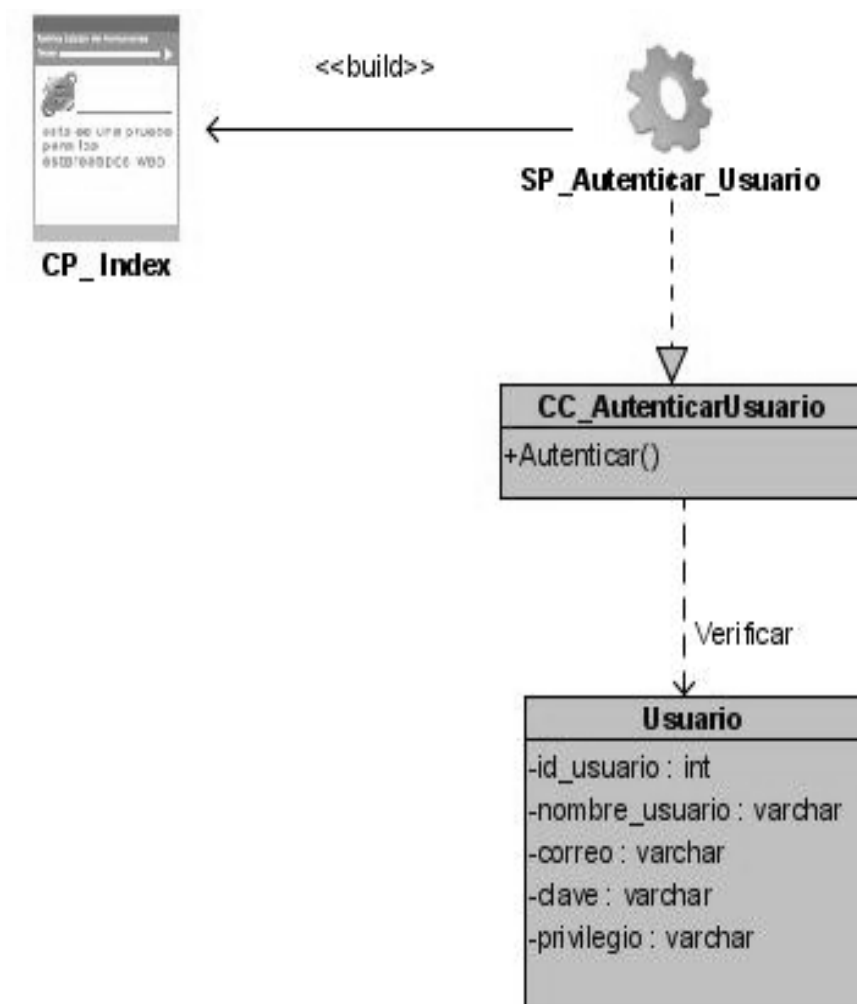


Fig. 4.25 Diagrama de clases del Diseño Web del CU Autenticar Usuario.

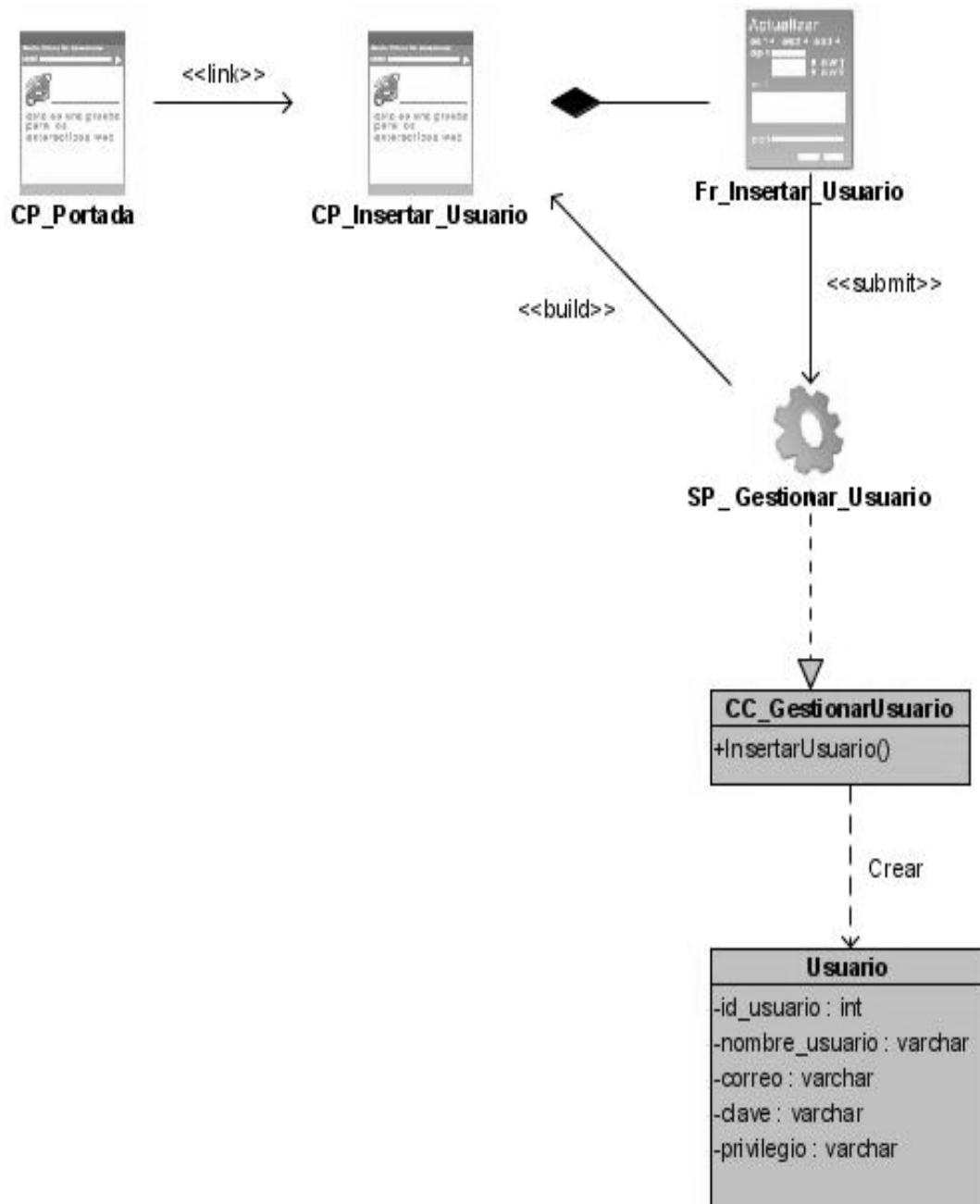


Fig. 4.26 Diagrama de clases del Diseño Web del CU Gestionar Usuario. Sección Insertar Usuario.

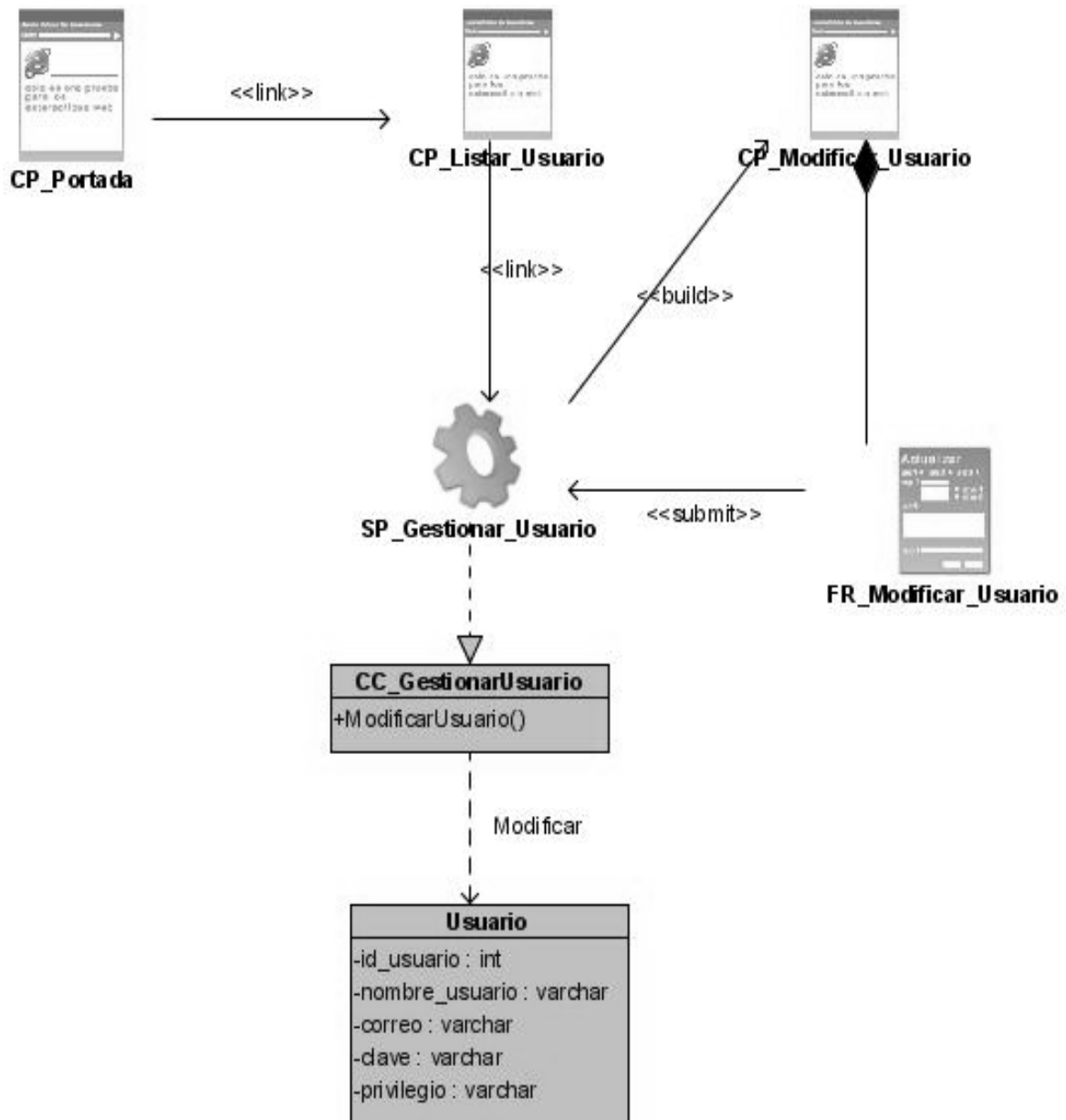


Fig. 4.27 Diagrama de clases del Diseño Web del CU Gestionar Usuario. Sección Modificar Usuario.

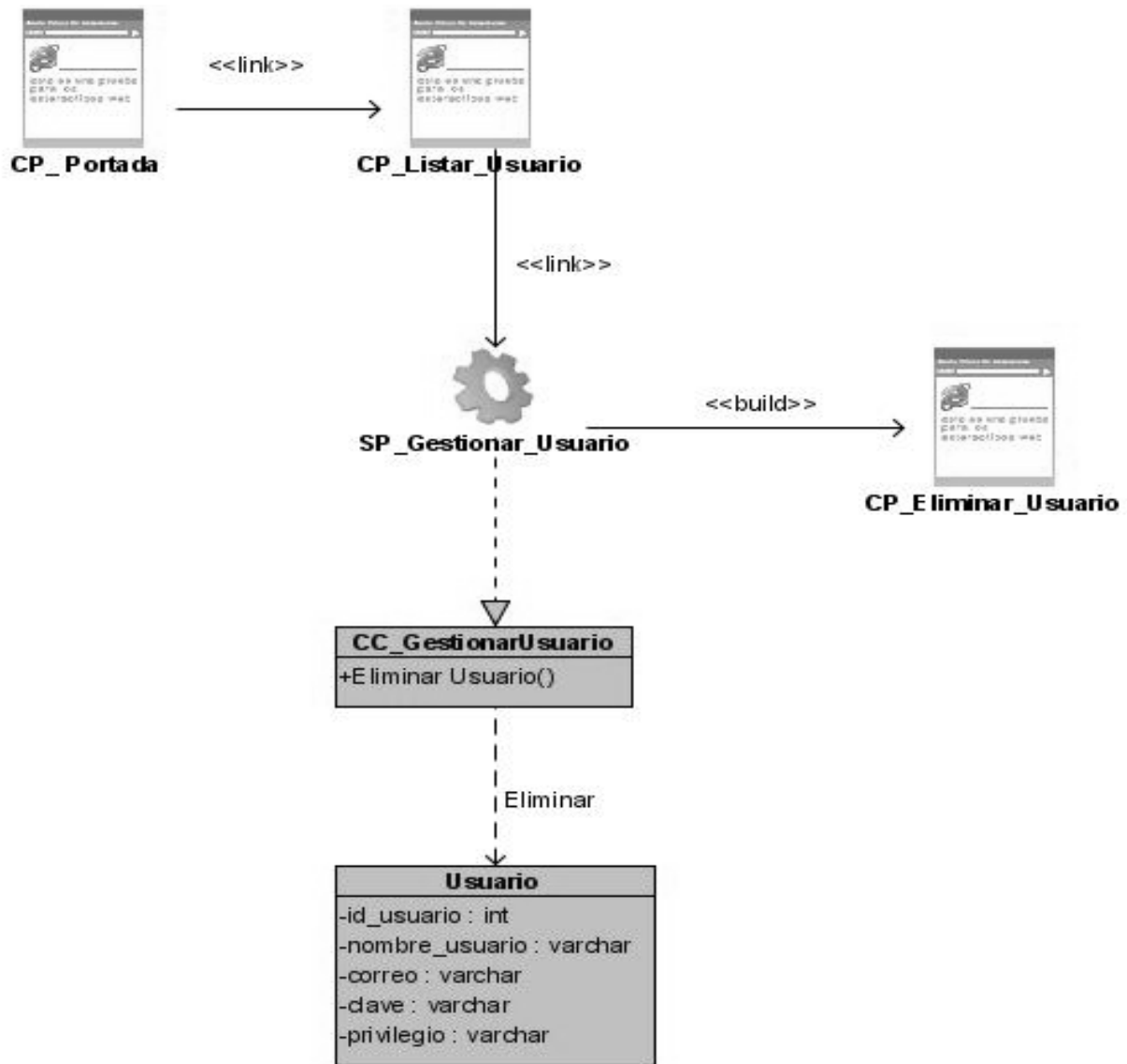


Fig. 4.28 Diagrama de clases del Diseño Web CU Gestionar Usuario. Sección Eliminar Usuario.

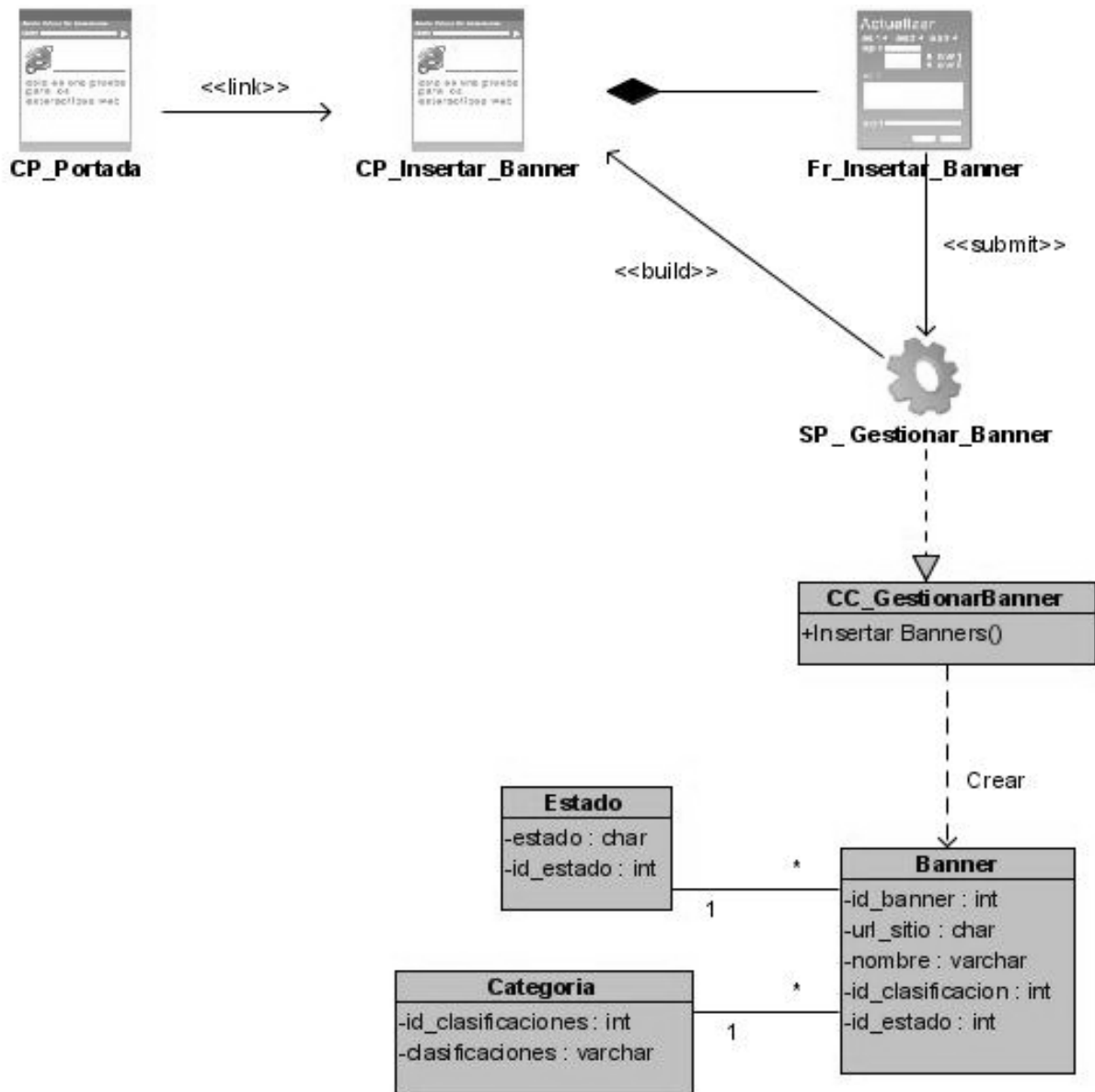


Fig. 4.29 Diagrama de clases del Diseño Web del CU Gestionar Banners. Sección Insertar Banner.

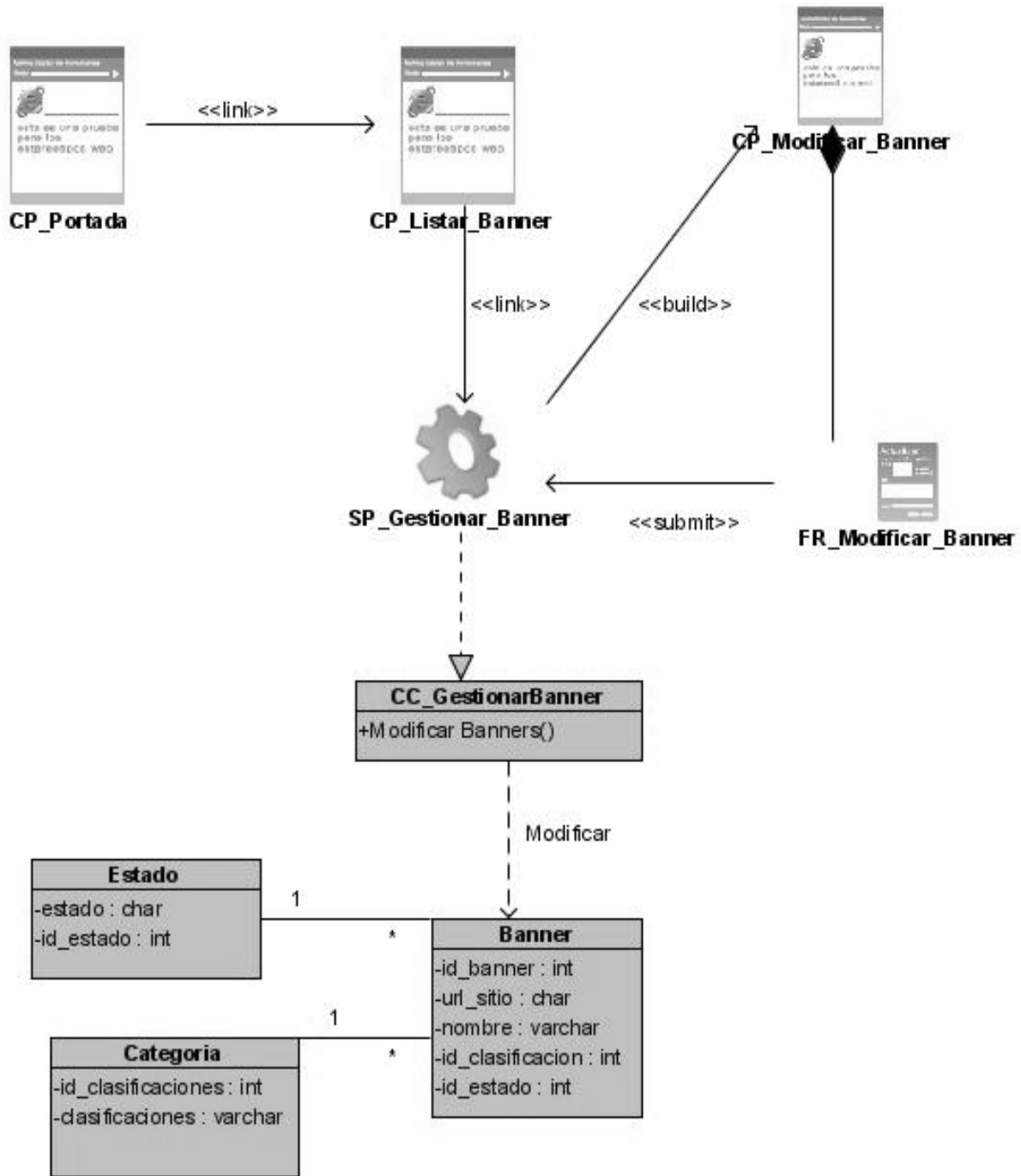


Fig. 4.30 Diagrama de clases del Diseño Web del CU Gestionar Banners. Sección Modificar Banner.

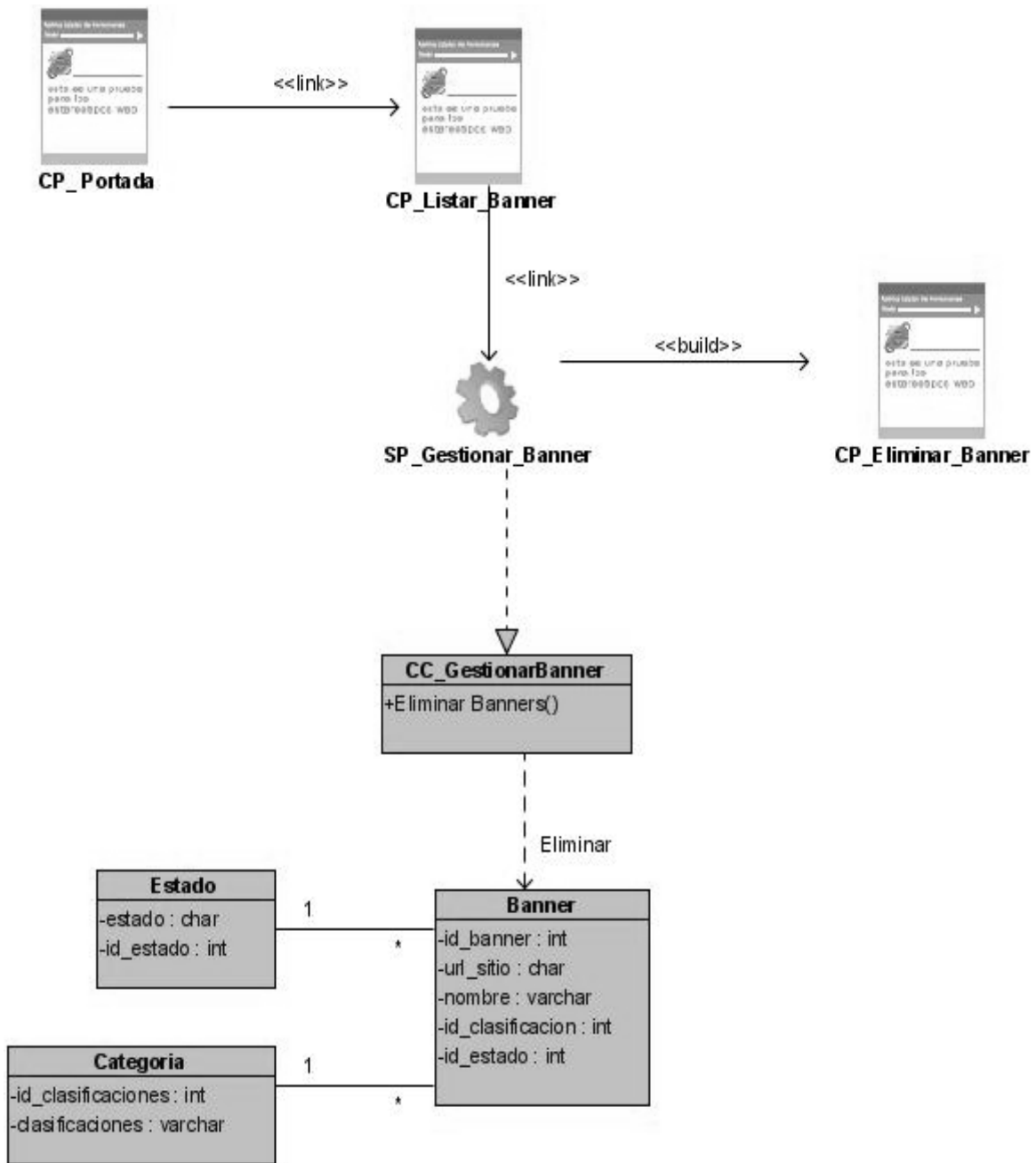


Fig. 4.31 Diagrama de clases del Diseño Web del CU Gestionar Banners. Sección Eliminar Banner.

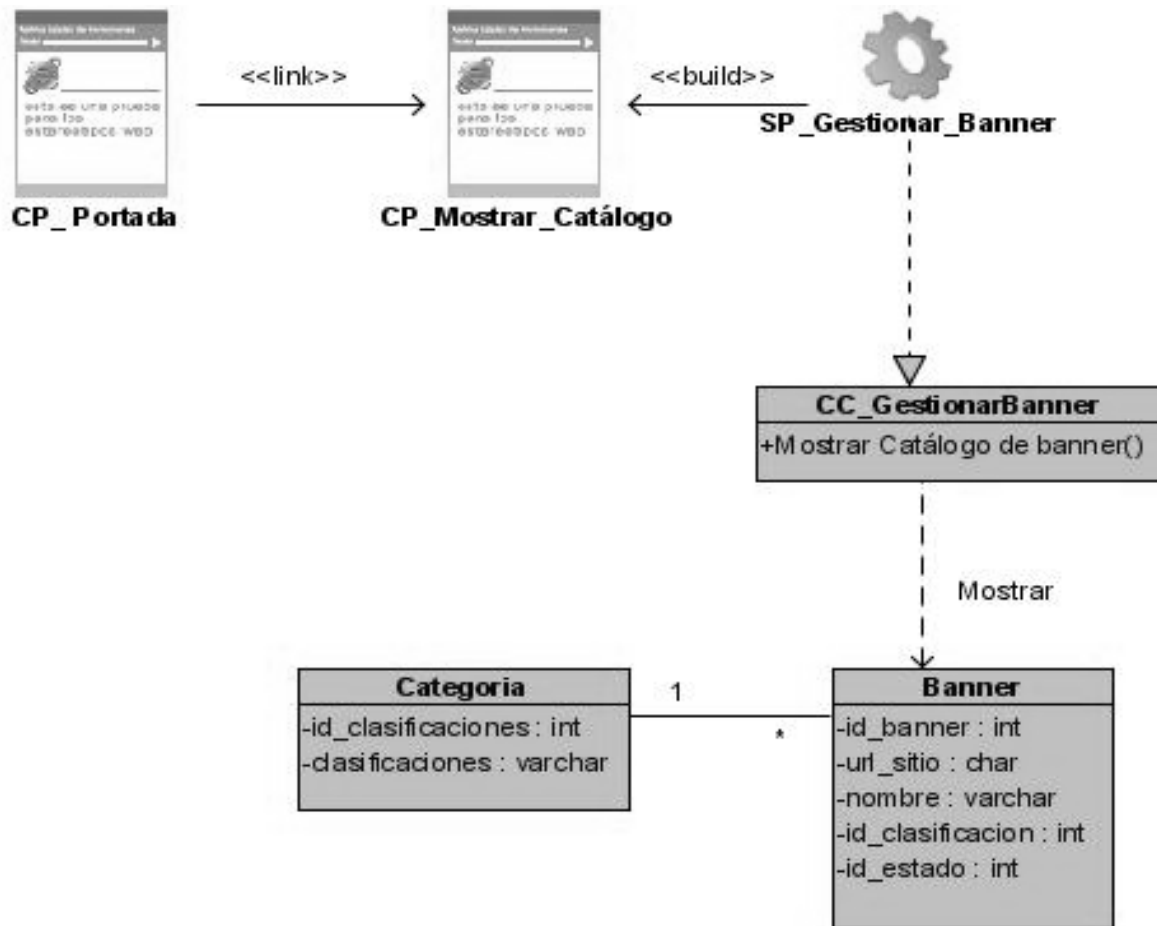


Fig. 4.32 Diagrama de clases del Diseño Web del CU Gestionar Banners. Sección Mostrar Catálogo de Banner por Categoría.

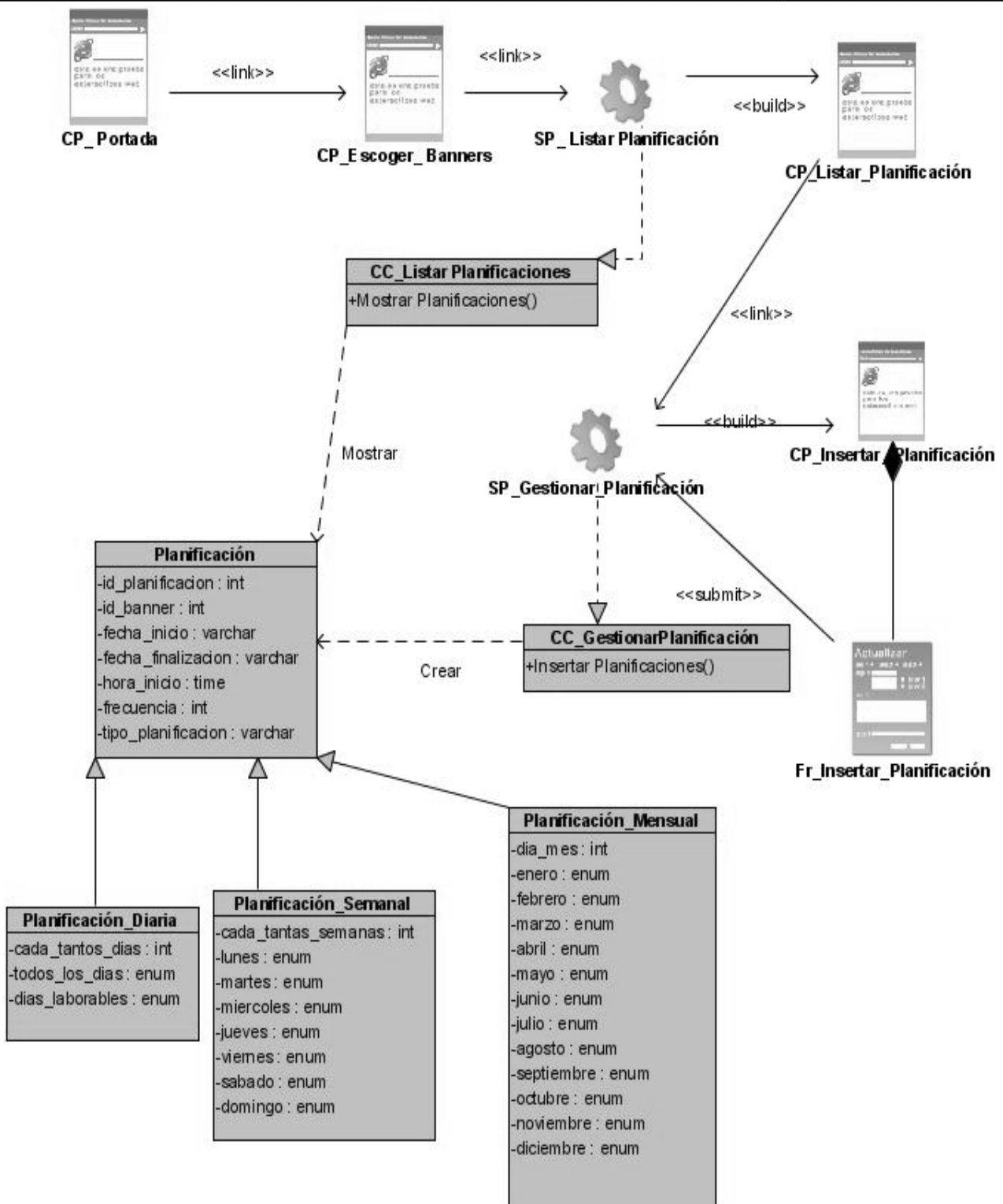


Fig. 4.33 Diagrama de clases del Diseño Web del CU Gestionar Planificación de Banner. Sección Insertar Planificación.

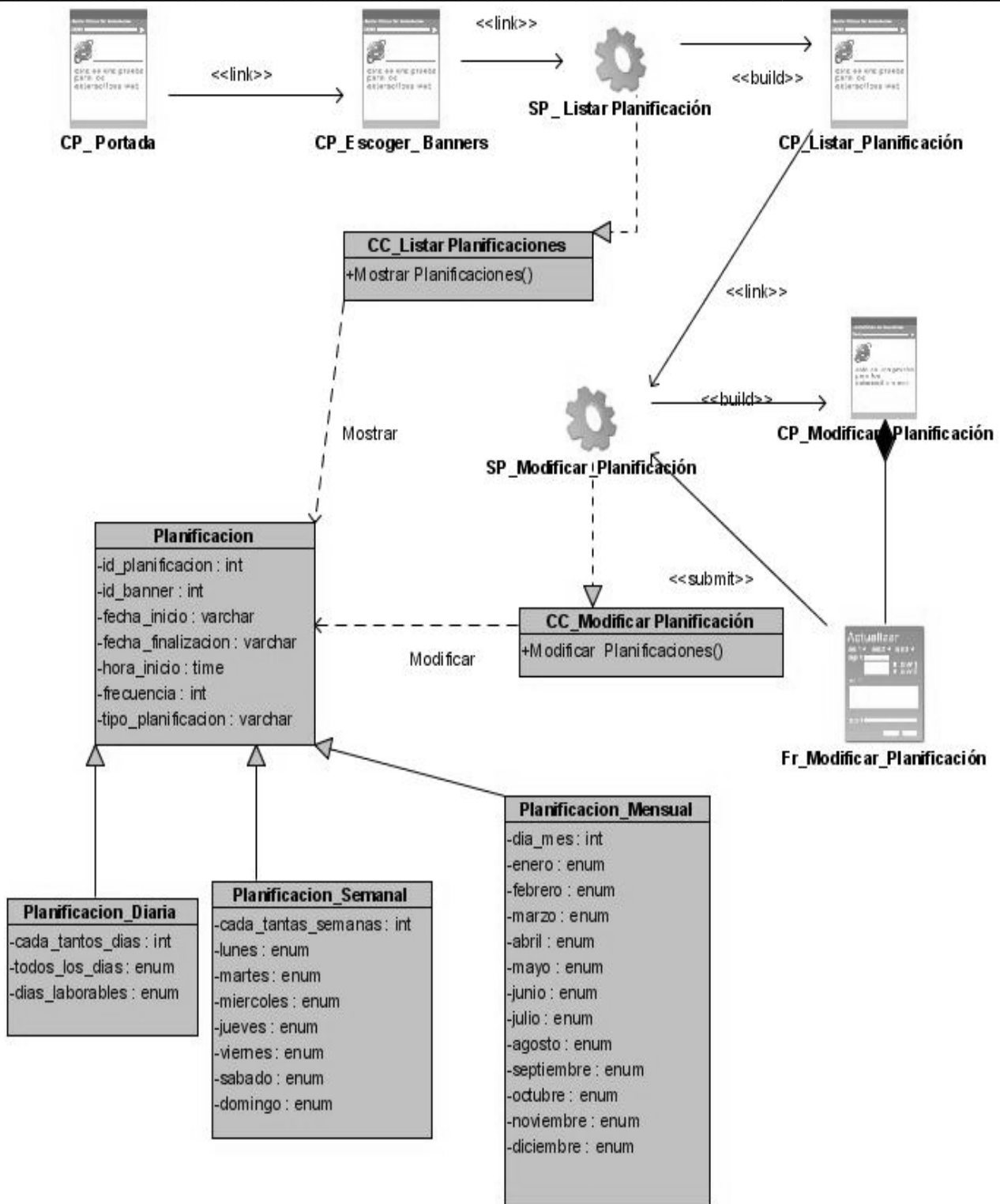


Fig. 4.34 Diagrama de clases del Diseño Web del CU Gestionar Planificación de Banner. Sección Modificar Planificación.

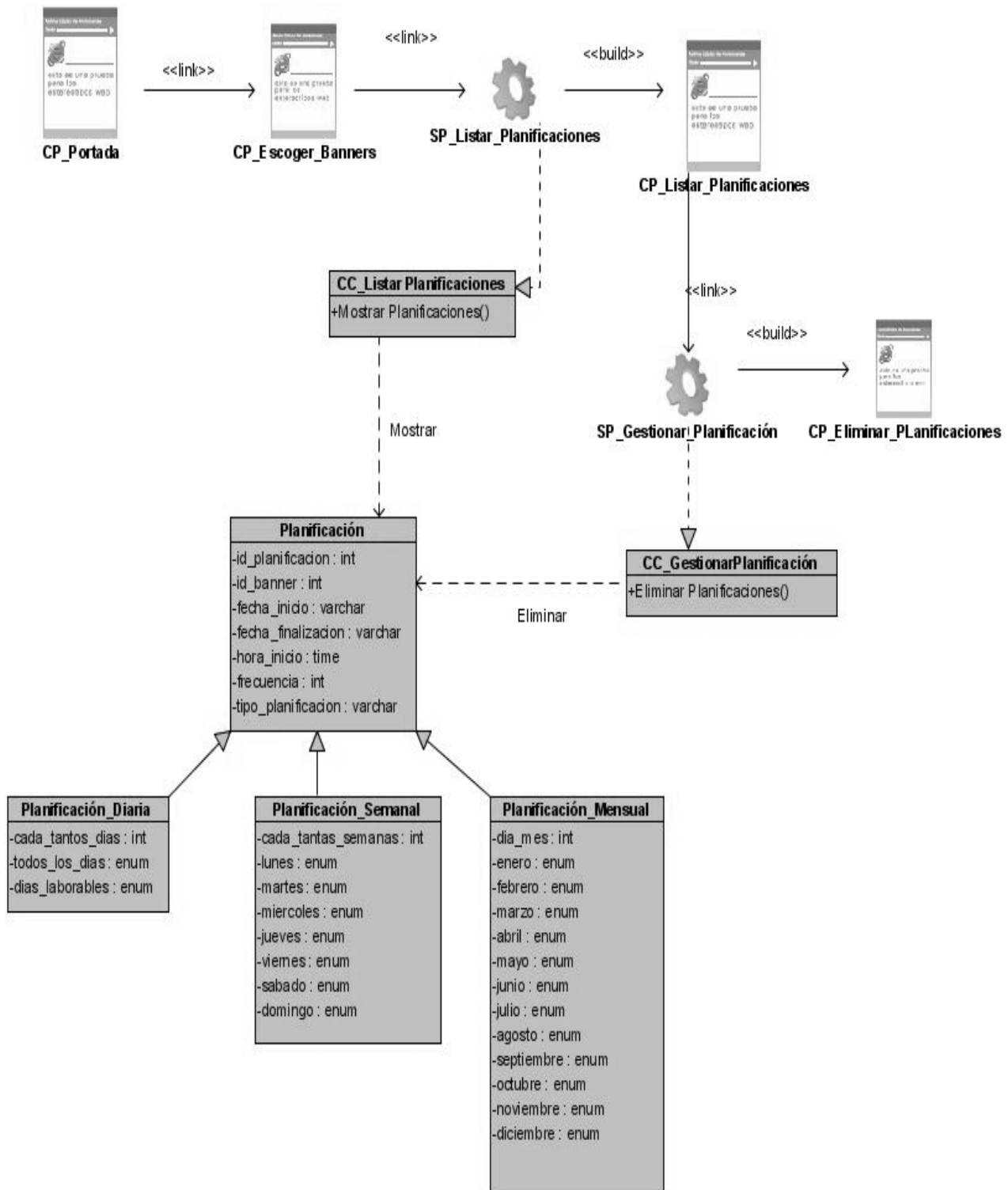


Fig. 4.35 Diagrama de clases del Diseño Web del CU Gestionar Planificación de Banner. Sección Eliminar Planificación.

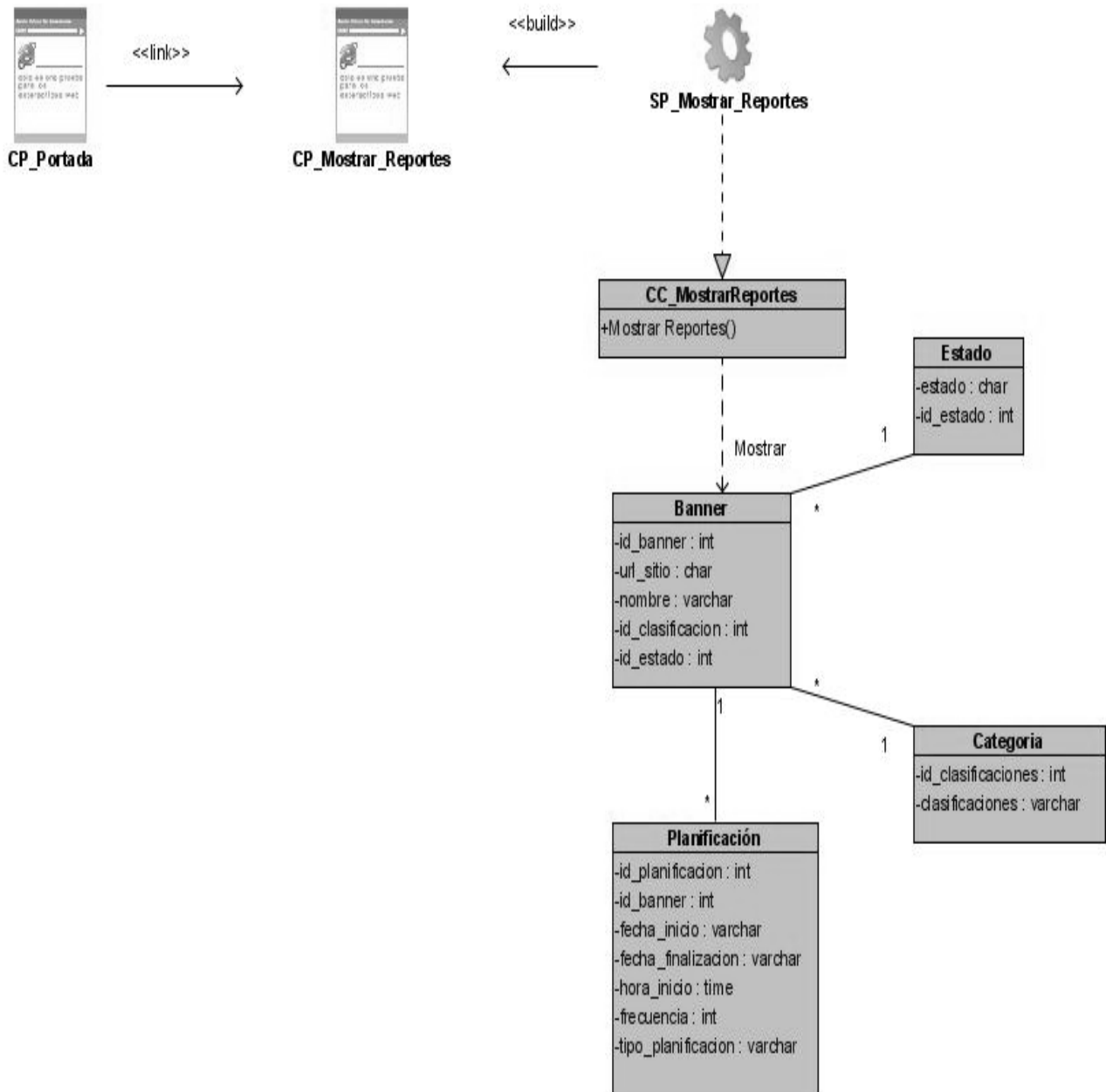


Fig. 4.36 Diagrama de clases del Diseño Web del CU Gestionar Reportes.

4.4 Principios del diseño.

4.4.1 Interfaz de usuario.

El sistema que se desarrolla representa una parte de los servicios Web que prestará la Intranet de la Universidad, a los usuarios y administradores de la red. Este hecho es determinante en la definición de la interfaz gráfica que se propone pues su diseño debe ser similar al estándar que tiene la Intranet, por lo que se utilizan tonalidades de azul y gris.

Todo el diseño está determinado fundamentalmente por el principio de la usabilidad, teniendo en cuenta que no se trata de un sitio Web, sino de una aplicación de trabajo donde el diseño tiene como principal propósito facilitar su uso, comprensión y navegación, por encima de ornamentos inútiles, aunque manteniendo pautas estéticas, orgánicas y agradables.

El interés general es mantener el diseño y la estructura del sitio lo más simple posible, la simplicidad es entendimiento del contenido, es facilidad para encontrar lo que se busca, es también velocidad de descarga, por eso se recomienda que las página no estén muy cargadas de imágenes.

4.5 Tratamiento de errores.

Una excepción es lanzada durante la ejecución del programa interrumpiendo el flujo normal de las sentencias. Permite de forma clara controlar los errores que ocurren en el momento de ejecución.

Cada formulario se encarga de la validación de sus datos para evitar errores de concepto. Y se utilizan mensajes de confirmación, para acciones que son irreversibles como es el caso de las eliminaciones.



Figura 4.37 – Mensaje de confirmación al eliminar los banners.

Presencia de errores en forma de mensaje de texto en la misma página donde se ejecuta la acción, de forma que el usuario pueda corregir el error fácilmente.



Figura 4.38 – Mensaje de error al insertar las contraseñas.

4.6 Diseño de la Base de Datos.

En este epígrafe se representa el diseño de la base de datos a través del diagrama de clases persistentes y el diagrama entidad relación.

4.6.1 Diagrama de clases persistentes de la base de datos.

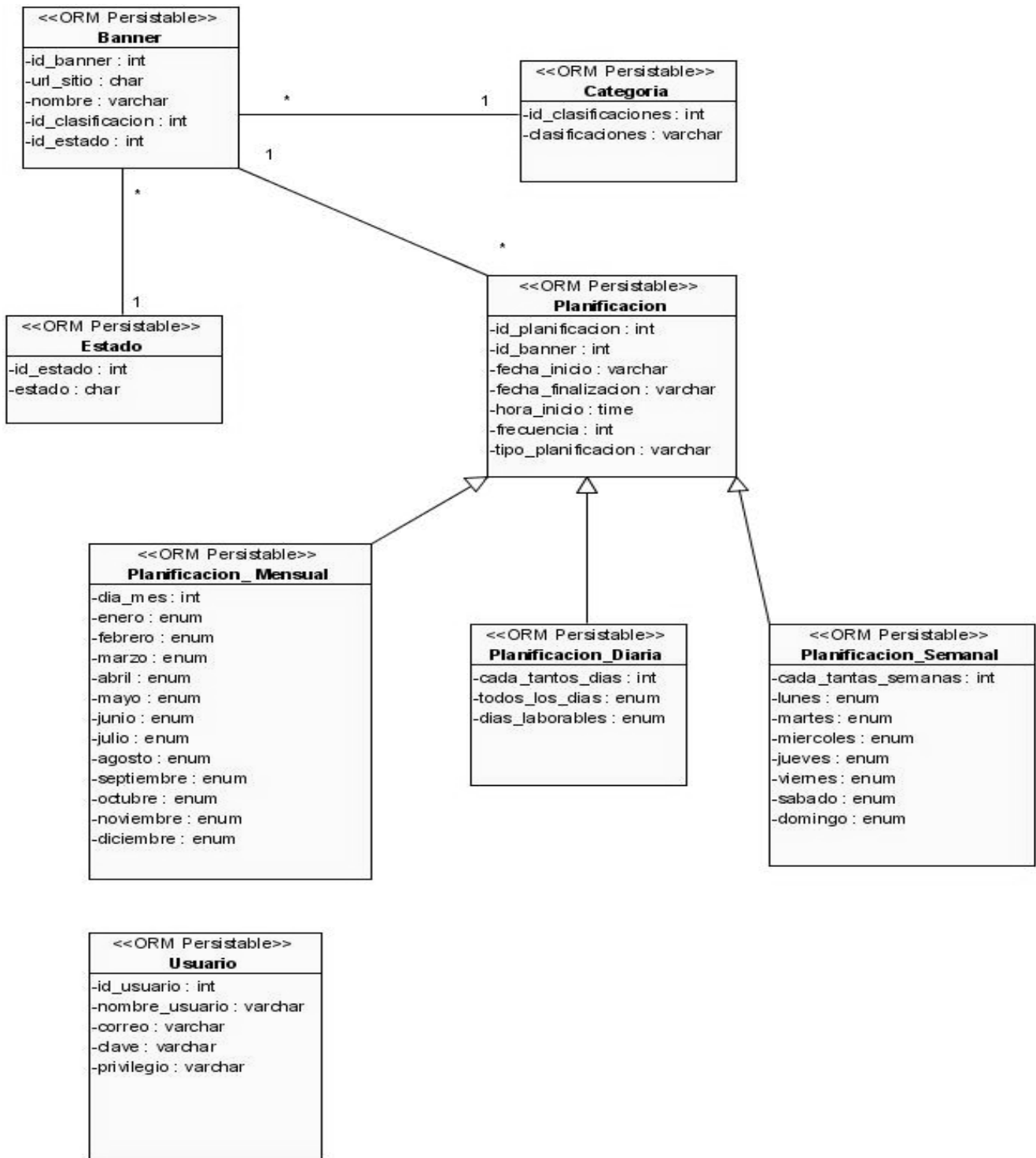


Figura 4.39 – Diagrama de clases persistentes de la base de datos.

4.6.2 Diagrama entidad relación de la base de datos.

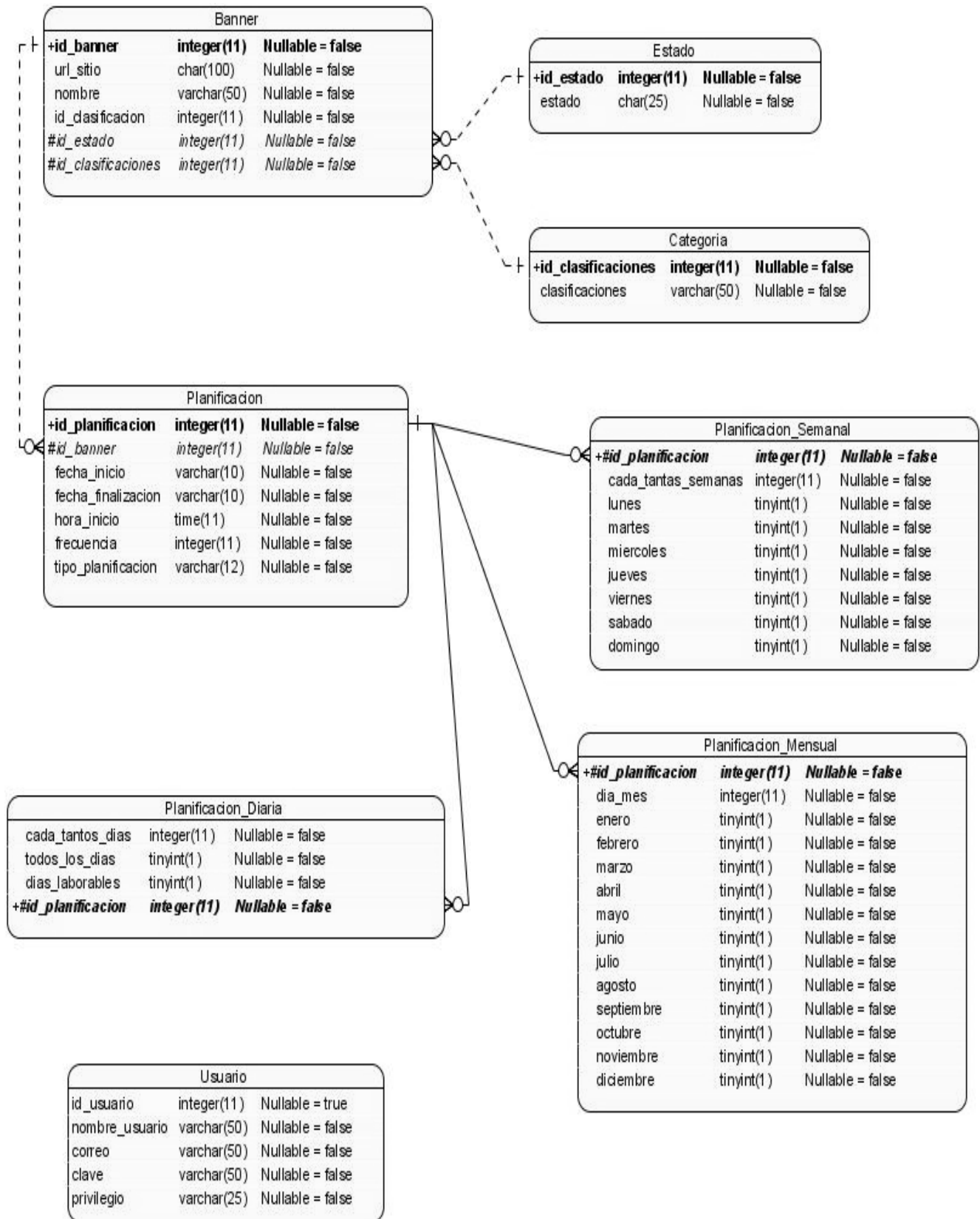


Figura 4.40 – Diagrama entidad relación de la base de datos.

4.6.3 Descripción de las tablas de la Base de Datos.

A continuación se describen cada una de de las tablas de la base de datos.

Nombre: Usuario		
Descripción: Esta tabla contiene los datos de los usuarios.		
Atributo	Tipo	Descripción
Id_usuario	int	Identificador de usuario. Valor auto numérico.
Nombre	varchar	Nombre del usuario.
Correo	varchar	Correo del usuario.
Clave	varchar	Contraseña para acceder al sistema.
Privilegio	varchar	Permisos que tienen definidos para el usuario.

Nombre: Banner		
Descripción: Esta tabla contiene los datos de los banners.		
Atributo	Tipo	Descripción
Id_banner	int	Identificador del banner. Valor auto numérico.
url_sitio	char	Url del Sitio donde se va a publicar el banner.
Nombre	varchar	Nombre del banner.
Id_clasificación	int	Identificador de la tabla categoría.
Id_estado	Int	Identificador de la tabla estado.

Nombre: Estado		
Descripción: Esta tabla contiene el estado de los banners.		
Atributo	Tipo	Descripción
Id_estado	int	Identificador del estado
estado	char	Estado del banner (Activo, No Activo).

Nombre: Categoría		
Descripción: Esta tabla contiene las categorías de los banners.		
Atributo	Tipo	Descripción
Id_clasificaciones	int	Identificador de la categoría. Valor auto numérico.
clasificaciones	varchar	Nombre de la categoría.

Nombre: Planificación		
Descripción: Esta tabla contiene las planificaciones de los banners.		
Atributo	Tipo	Descripción
Id_planificación	int	Identificador de la planificación. Valor auto numérico.
Id_banner	int	Identificador de la tabla banner.
fecha_inicio	varchar	Fecha que se va a publicar el banner en el sitio.
fecha_finalización	varchar	Fecha que se va a retirar el banner del sitio.
hora_inicio	time	Hora de empezar la publicación.
frecuencia	int	Frecuencia por minutos que el banner se va a publicar.
tipo_planificación	varchar	Indica el tipo de planificación (Diaria, Semanal, Mensual).

Nombre: Planificación_Semanal		
Descripción: Esta tabla contiene las planificaciones semanales, es una especialización de la tabla Planificación.		
Atributo	Tipo	Descripción
cada_tantas_semanas	int	Cantidad de semanas que va estar publicado el banner.
lunes	enum	Indica los días de la semana que va estar publicado.
martes	enum	
miércoles	enum	
jueves	enum	
viernes	enum	
sábado	enum	
domingo	enum	

Nombre: Planificación_Diaria		
Descripción: Esta tabla contiene las planificaciones diarias, es una especialización de la tabla Planificación.		
Atributo	Tipo	Descripción
cada_tantos_días	int	Cantidad de días que va a estar publicado el banner.
todos_los_días	enum	Indica que se publicaran todos días.
días_laborables	enum	Indica que se publicaran solamente los días laborables.

Nombre: Planificación_Mensual		
Descripción: Esta tabla contiene las planificaciones mensuales, es una especialización de la tabla Planificación.		
Atributo	Tipo	Descripción
día_mes	int	Día de los meses que se va a publicar el banner.
enero	enum	Indica los meses que va estar publicado.
febrero	enum	
marzo	enum	
abril	enum	
mayo	enum	
junio	enum	
julio	enum	
agosto	enum	
septiembre	enum	
octubre	enum	
noviembre	enum	
diciembre	enum	

4. 7 Conclusiones.

En este capítulo se ha llevado a cabo los diagramas de interacción, la descripción de las clases y demás elementos necesarios para la implementación. Se definieron, cuáles son las clases que serán persistentes, luego, a partir de esto, se construyó el modelo de datos. Se expusieron las pautas seguidas para el diseño de la interfaz, y se explicó cómo se llevó a cabo el tratamiento de errores.

Capítulo 5: Implementación y Prueba.

5.1 Introducción.

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. Mediante el modelo de componentes se presenta la visa estática del sistema, además se muestra el modelo de prueba, en el cual se describen los casos de prueba de integración por cada caso de uso.

5.2 Diagrama de despliegue.

Los diagramas de despliegue se utilizan para razonar sobre la topología de procesadores y dispositivos sobre los que se ejecuta el software. Se define un diagrama de despliegue como un diagrama que muestra la configuración de los nodos que participan en la ejecución y de los componentes que residen en ellos. Gráficamente, un diagrama de despliegue es una colección de nodos y arcos.

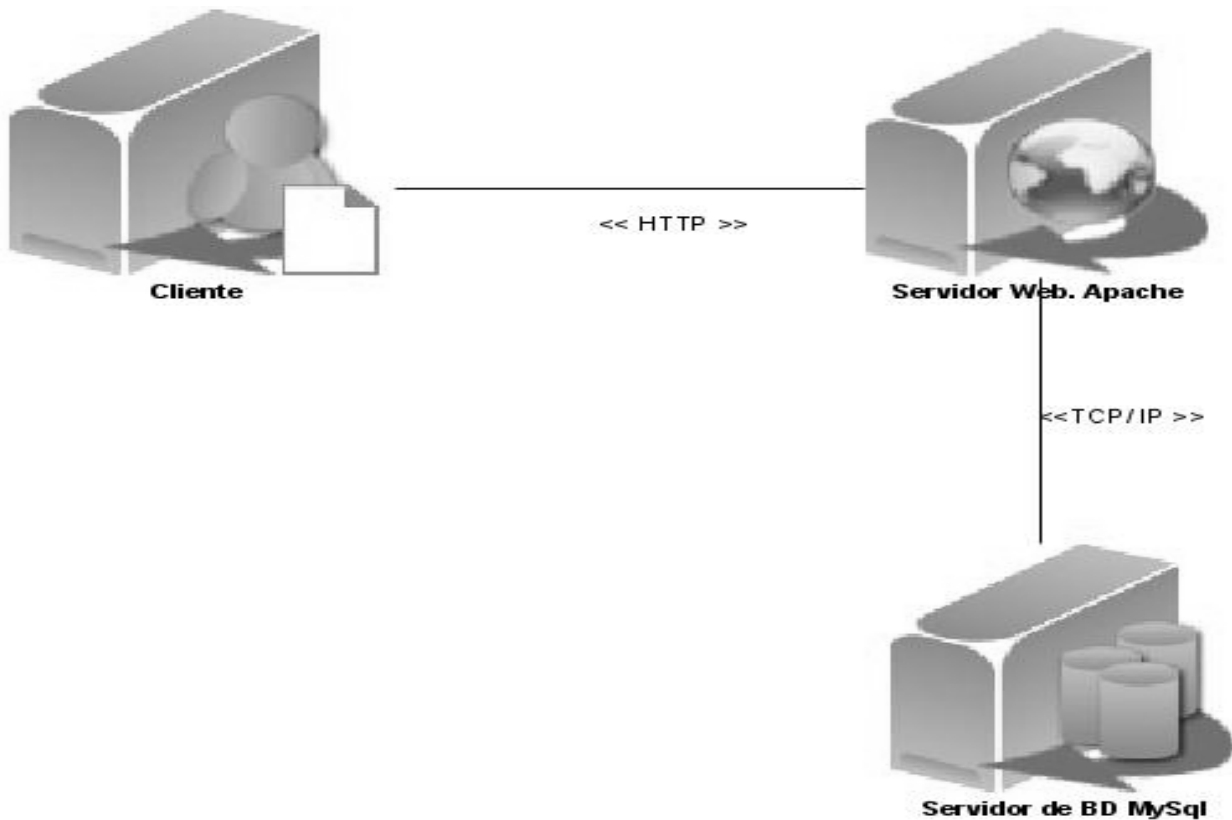


Figura 5.1 – Diagrama de despliegue.

5.3 Diagramas de componentes.

Los diagramas de componentes se utilizan para modelar la vista de implementación del sistema. Esto implica modelar las cosas físicas que existen en un nodo, tales como ejecutables, bibliotecas, tablas, archivos y documentos.

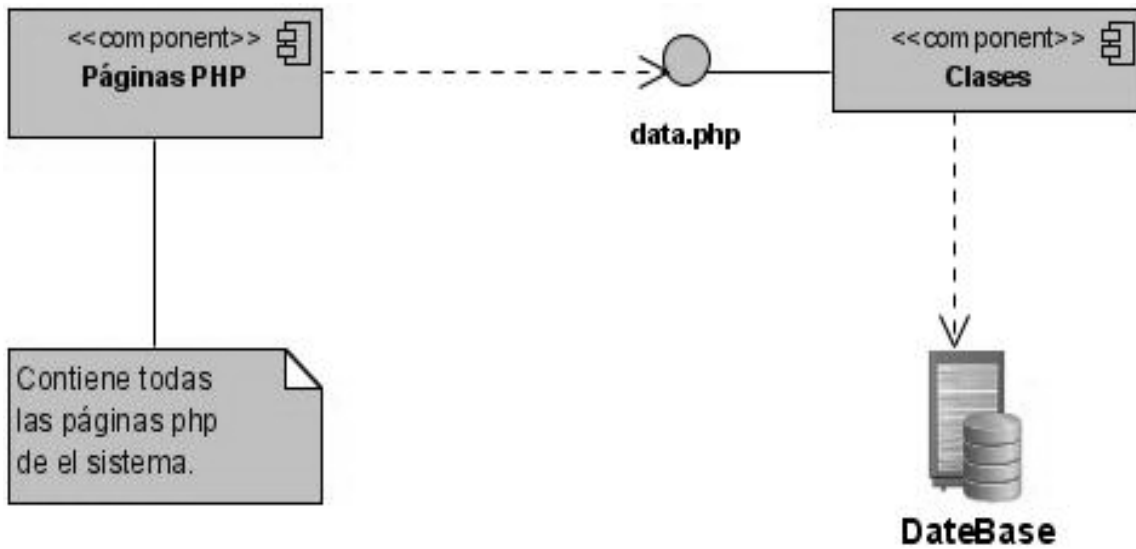


Figura 5.2 – Diagrama de Componentes de Bases de Datos.

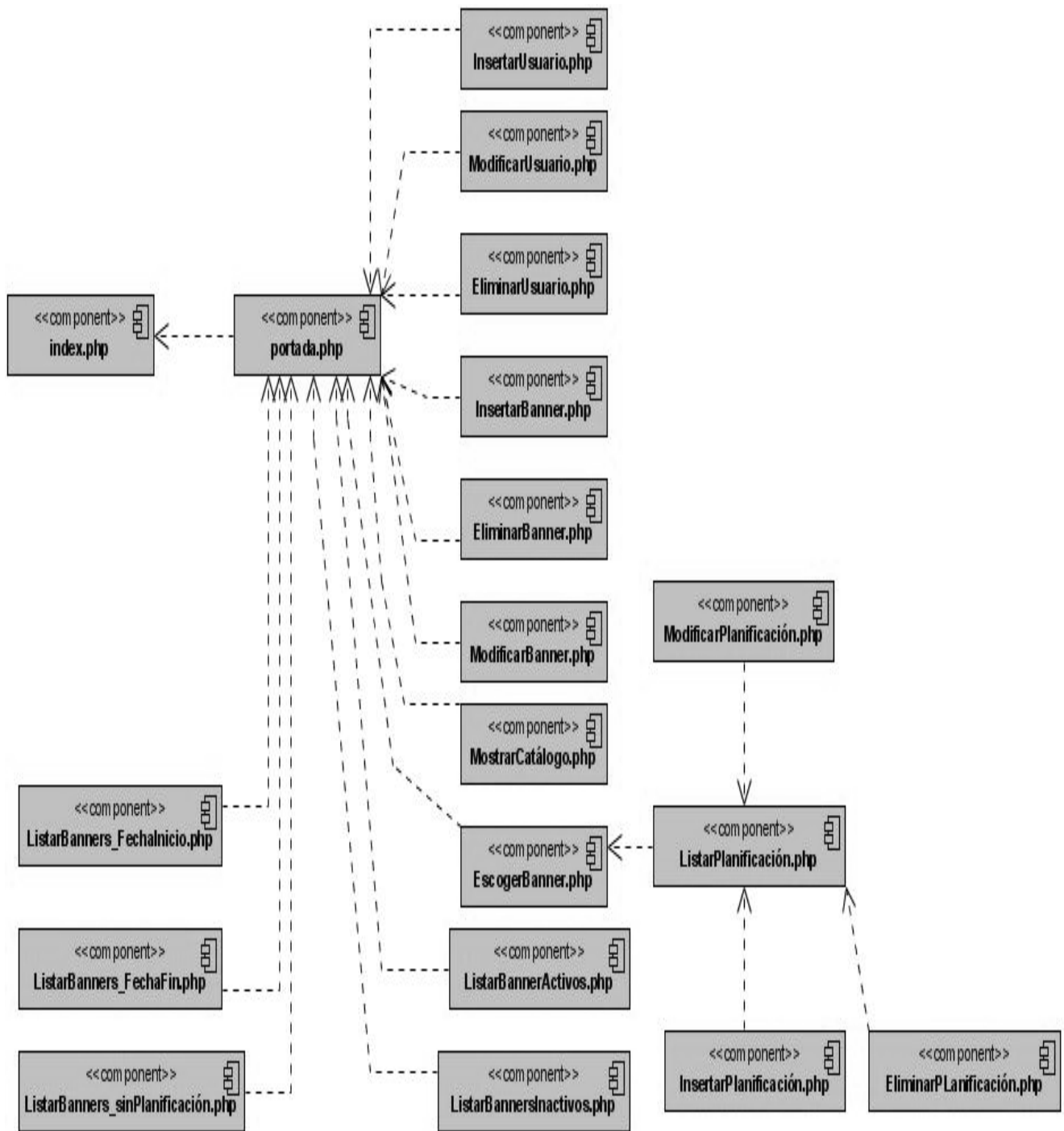


Figura 5.3 – Diagrama de Componentes de las relaciones de las páginas PHP.

5.4 Modelo de prueba.

Nombre del caso de uso: Autenticar Usuario

Entrada	Resultados	Condiciones
El usuario introduce un nombre de usuario y contraseña incorrecta.	El sistema muestra un mensaje: El usuario o el password no existen.	
El usuario introduce correctamente el nombre de usuario, pero incorrectamente la contraseña.	El sistema muestra un mensaje: Password incorrecto.	
El usuario introduce correctamente su nombre de usuario y contraseña.	El sistema le da acceso al sitio de acuerdo a sus privilegios.	



Figura 5.4 – Mensaje de error al usuario autenticarse.



Figura 5.5 – Mensaje de error al usuario autenticarse.

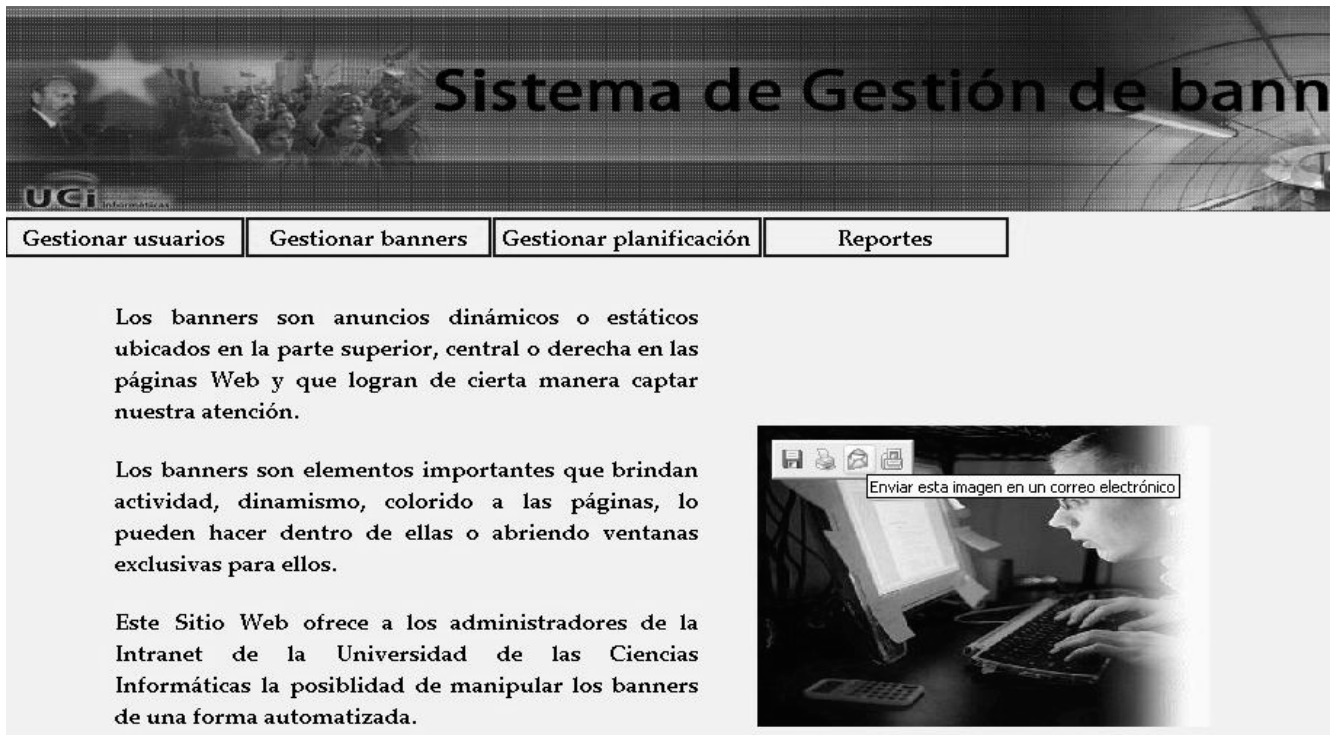


Figura 5.6 – El usuario autenticado tiene acceso al sitio de acuerdo a su privilegio.

Nombre del caso de uso: Gestionar Usuario

Entrada	Resultados	Condiciones
El Administrador del sistema cuando va a insertar o modificar una cuenta de un usuario deja campos en blanco.	El sistema muestra un mensaje: Debe llenar todos los campos.	
El Administrador cuando esta modificando o insertando una cuenta de usuario confirmó incorrectamente la clave.	El sistema muestra un mensaje: Las claves deben ser las mismas...rectifíquelas.	
El Administrador cuando esta modificando o insertando una cuenta de usuario y escribe un nombre de usuario que ya esta registrado.	El sistema muestra un mensaje: Ya existe un usuario con dicho nombre... Cámbielo.	
El administrador desea eliminar una cuenta de un usuario.	El sistema muestra un mensaje: Realmente desea eliminar los usuarios seleccionados.	



Figura 5.7 – Mensaje de error al insertar o modificar un usuario.

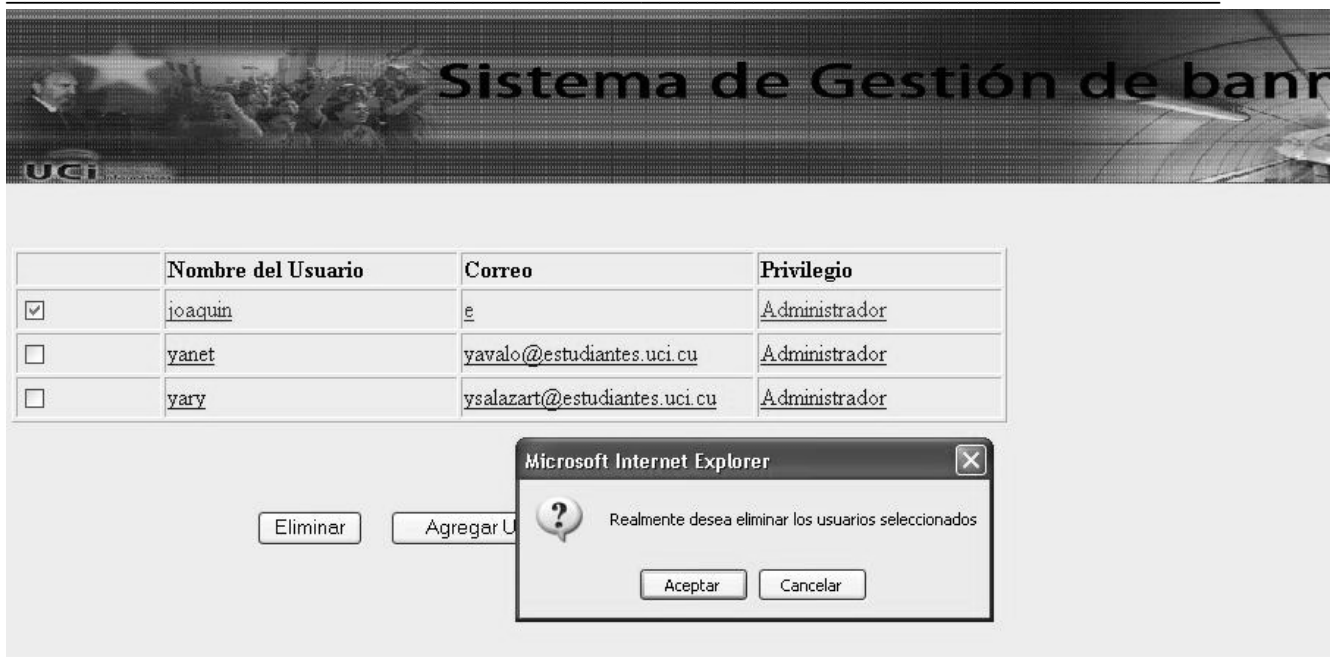


Figura 5.8 – Mensaje de confirmación al eliminar un usuario.



Figura 5.9 – Mensaje de error al insertar o modificar un usuario.

Nombre del caso de uso: Gestionar Banners

Entrada	Resultados	Condiciones
El Administrador cuando esta modificando o insertando un banner deja	El sistema muestra un mensaje: Debe llenar todos los campos.	

campos en blanco.		
El administrador desea eliminar un banner que caducó.	El sistema muestra un mensaje: Realmente desea eliminar los banners seleccionados.	
El administrador da click en el vínculo Mostrar Catálogo.	El sistema muestra un listado de banners por la categoría que el administrador desee.	



Figura 5.10 – Mensaje de error al insertar o modificar un banner.



Figura 5.11 – Muestra los banners de acuerdo a la categoría que buscas.

Nombre del caso de uso: Gestionar Planificación de Banners

Entrada	Resultados	Condiciones
El administrador escoge el banner que desea modificar o insertar una planificación.	El sistema muestra un formulario con todos los campos que requiere la planificación inicializada.	
El administrador desea eliminar la planificación de un banner.	El sistema muestra un mensaje: Realmente desea eliminar las planificaciones seleccionadas.	

PLANIFICACION

fecha de inicio dia 1 ▼ mes Enero ▼ año 2007 ▼

hora de inicio 0 ▼ : 0 ▼

frecuencia 1 minutos

fecha de finalizacion dia 1 ▼ mes Enero ▼ año 2007 ▼

Tipo de planificacion Diariamente Semanalmente Mensualmente

Cada 1 semanas

Seleccione el día o los días de la semana.

Lunes Jueves Domingo

Martes Viernes

Miércoles Sábado

Figura 5.12 – Página para insertar o modificar una planificación.

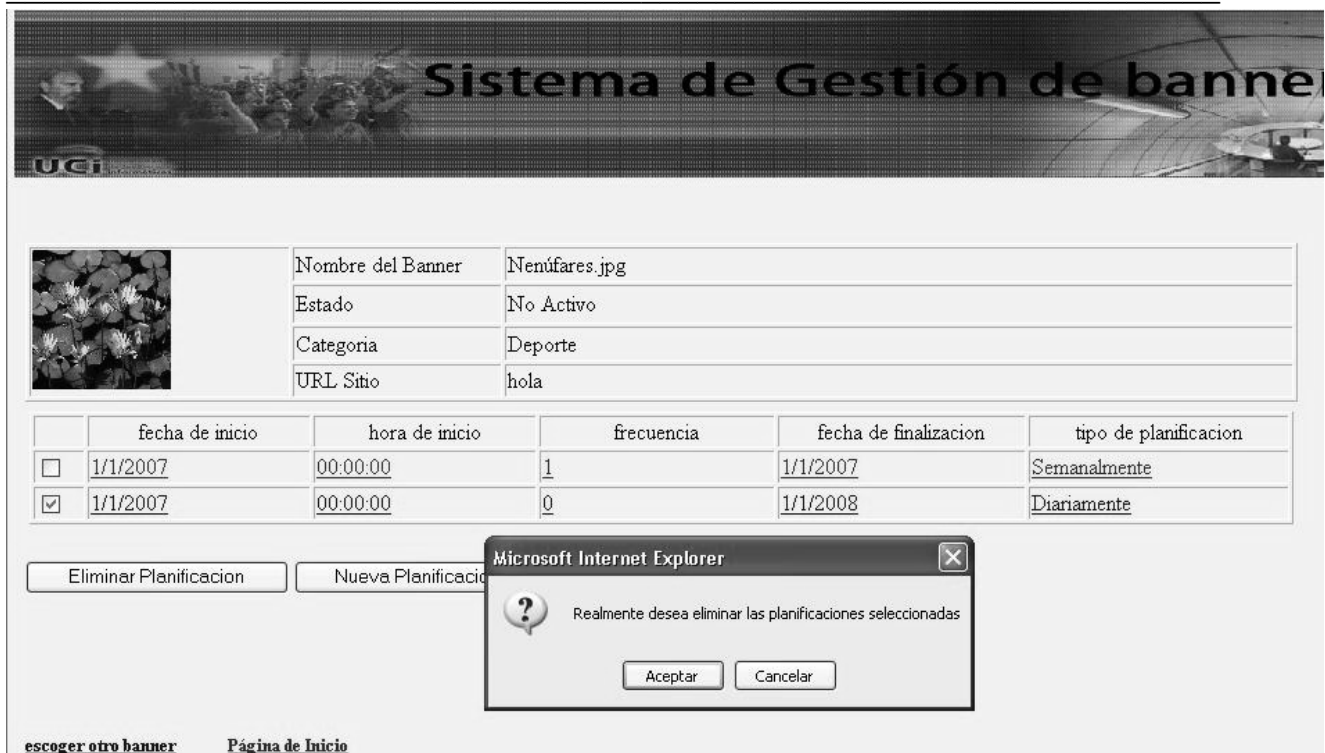


Figura 5.13 – Mensaje de confirmación al eliminar una planificación.

Nombre del caso de uso: Gestionar Reportes

Entrada	Resultados	Condiciones
El administrador da click en el vínculo Listar Banners Activos.	El sistema muestra un listado de los banners que su estado sea activo.	
El administrador da click en el vínculo Listar Banners Inactivos.	El sistema muestra un listado de los banners que su estado sea no activo.	
El administrador da click en el vínculo Listar Banners con la fecha de publicación	El sistema muestra un listado de los banners con su fecha de publicación.	
El administrador da click en el vínculo Listar Banners con la fecha de finalización.	El sistema muestra un listado de los banners con su fecha de finalización.	
El administrador da click en el vínculo Listar Banners sin planificación.	El sistema muestra un listado de los banners que no tienen ninguna planificación.	



Figura 5.14 – Muestra los banners que su estado sea activo.

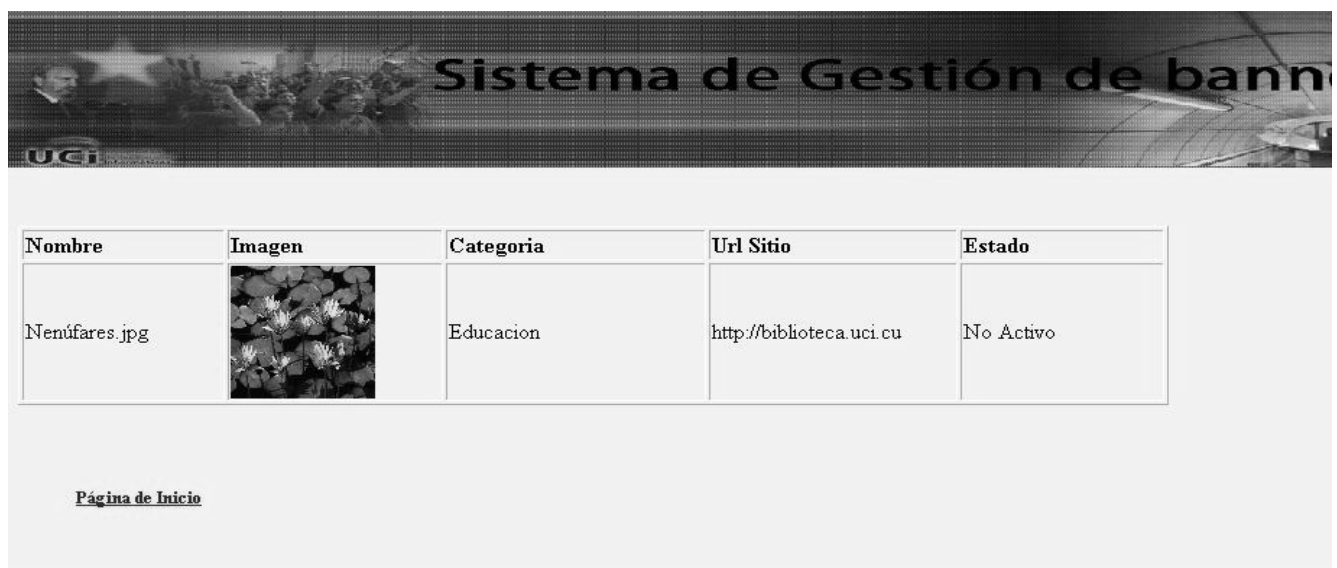


Figura 5.15 – Muestra los banners que su estado sea no activo.



Nombre	Imagen	Categoría	Url Sitio	Estado	Fecha de publicación	Tipo de Planificación
Puesta de sol.jpg		Deporte	http://intranet.uci.cu	Activo	3/6/2007	Diariamente
Nenúfares.jpg		Educacion	http://biblioteca.uci.cu	No Activo	22/1/2007	Semanalmente

Figura 5.16 – Muestra los banners que con su fecha de publicación.





Nombre	Imagen	Categoría	Url Sitio	Estado	Fecha de finalización	Tipo de Planificación
Puesta de sol.jpg		Deporte	http://intranet.uci.cu	Activo	28/7/2007	Diariamente
Nenúfares.jpg		Educacion	http://biblioteca.uci.cu	No Activo	23/7/2007	Semanalmente

Figura 5.17 – Muestra los banners con su fecha de finalización.



Figura 5.18 – Muestra los banners que no tienen planificación.

5.5 Conclusiones.

En este capítulo se mostraron varias vistas para llevar a cabo el proceso de implementación del sistema. Igualmente se identificaron otras funcionalidades que se pueden tener en cuenta para futuras versiones del sistema. Se especificaron los protocolos de comunicación, mediante el diagrama de despliegue. Se representaron las dependencias entre los componentes del software a través del diagrama de componentes y por último se realizó la descripción de los casos de prueba de integración de cada uno de los casos de uso del sistema.

Conclusiones

Una vez concluido satisfactoriamente el siguiente trabajo, se dieron cumplimiento a los objetivos planteados, obteniéndose los siguientes resultados:

- La realización de una Base de Datos, que se encarga de guardar toda la información de la solicitud de los banners.
- La implementación del Sistema Automatizado de Banners para el módulo de los administradores de la Intranet, que les proporciona el almacenamiento y control de los banners de una forma más conveniente y segura.

Con el funcionamiento de este sistema se logra un aumento de la calidad de la Intranet de la Universidad de las Ciencias Informáticas, solucionando una de las deficiencias actualmente existentes en este sitio Web, que es la administración de los banners que se publican en la misma.

El sistema desarrollado guarda en una base de datos todos los banners clasificados por categoría, logrando así una magnífica organización de estos, lo cual facilita enormemente el trabajo de los administradores de la intranet a la hora de realizar la planificación para publicarlos, ya que el sistema posee una funcionalidad para realizar dicha planificación, lo cual garantiza que los banners no se conserven tanto tiempo en la Intranet sin ser renovados, y el usuario que acceda a este sitio no pierda el interés por la información que en ellos aparece. Además el sistema implementa el control de la seguridad, obligando a los usuarios a realizar una autenticación para acceder, permitiendo solamente el acceso a las personas autorizadas y definiendo por roles los permisos de navegación y privilegios dentro del mismo.

Recomendaciones

Con vistas al desarrollo futuro de este proyecto se listan a continuación una serie de recomendaciones:

- Dar continuidad a la implementación del sistema del módulo solicitante, que les permita a los usuarios insertar la solicitud de los banners en el sistema.
- La planificación de los banners en el sistema actual se realiza uno a uno obligatoriamente, lo cual podría mejorarse, dando la posibilidad de que se pueda realizar la planificación de varios al mismo tiempo.
- Continuar con la investigación para la ampliación, modificación, mejora y construcción de nuevas versiones de este sistema.

Referencias Bibliográficas

1. **Alvarez, Miguel Angel. 2006.** Zend Studio. 2006.
<http://www.desarrolloweb.com/articulos/1178.php>
2. **Anónimo. 2004.** ¿Qué es ASP? 2004.
<http://www.aspfacil.com/articulos/278001.asp>
3. —. **2007.** Cursos de Perl. 2007.
http://www.wikilearning.com/recursos_perl-wkk-2563.htm
4. —. **2006.** Qué es Javascript. 2006.
<http://www.desarrolloweb.com/articulos/25.php>
5. —. **2004.** Sistema de intercambio de Banners Servicio. 24 de marzo de 2004.
<http://www.webservicio.com/banners>
6. —. **2004.** Sitio web de bannerlandia. 24 de marzo de 2004.
<http://www.bannerlandia.com/promociona/intercambio-de-banners/>
7. **Arambillete, Wilman. 2002.** Introducción a las Java Server Pages. 18 de junio de 2002.
<http://www.webexperto.com/articulos/art/57/introduccion-a-las-java-server-pages/>
8. **Castellanos, Argenis González. 2005.** Comparación entre Sistemas de Gestión de Bases de Datos (SGBD) bajo licenciamiento libre y comercial. 2005.
<http://www.ilustrados.com/documentos/sghbd.pdf>
9. **Ciberaula. 2006.** Introducción a Dreamweaver 3. 2006.
<http://www.programacion.net/html/tutorial/dreamweaver/2/>
10. **Flores, J. Tomás Nogales. 2006.** Introducción a XSLT (Extensible Stylesheet Language - Transformations). 10 de enero de 2006.
<http://www.bib.uc3m.es/~nogales/cursos/xslt.html>
11. **Pey, Xavier. 2001.** ASP.NET. 11 de junio de 2001.
http://www.netveloper.com/imprimir.aspx?IDC=39_0
12. **Roco, Pier Angeli Zorzini. 2005.** La publicidad. febrero de 2005.

Referencias Bibliográficas

<http://www.monografias.com/trabajos20/presupuesto-publicidad/presupuesto-publicidad.shtml>

13. **Rodríguez, Toni Soto. 2002.** GaliciaCity. 11 de octubre de 2002.

<http://www.galiciacity.com/presenta.shtml>

14. **S., Christian Van Der Hest. 2001.** ¿Qué es el PHP? 23 de mayo de 2001.

<http://www.maestrosdelweb.com/editorial/phpintro/>

15. **Sanchez, María A. Mendoza. 2004.** Metodologías De Desarrollo De Software. 2004.

http://www.informatizate.net/articulos/metodologias_de_desarrollo_de_software_07062004.html

1. **Alvarez, Miguel Angel. 2006.** Adobe Golive. 2006.
<http://www.desarrolloweb.com/articulos/343.php>
2. **Alvarez, Miguel Angel. 2006.** Zend Studio. 2006.
<http://www.desarrolloweb.com/articulos/1178.php>
3. **Anónimo. 2007.** Cursos de Perl. 2007.
http://www.wikilearning.com/recursos_perl-wkk-2563.htm
4. —. **2006.** Qué es Javascript. 2006.
<http://www.desarrolloweb.com/articulos/25.php>
5. —. **2006.** Quanta Plus 3.5.1. 1 de febrero de 2006
<http://quanta-plus.softonic.com/linux>
6. —. **2004.** Sistema de intercambio de Banners Servicio. 24 de marzo de 2004.
<http://www.webservicio.com/banners>
7. —. **2004.** Sitio web de bannerlandia. 24 de marzo de 2004.
<http://www.bannerlandia.com/promociona/intercambio-de-banners/>
8. —. **2004.** ¿Qué es ASP? 2004.
<http://www.aspfacil.com/articulos/278001.asp>
9. **Arambillete, Wilman. 2002.** Introducción a las Java Server Pages. 18 de junio de 2002.
<http://www.webexperto.com/articulos/art/57/introduccion-a-las-java-server-pages/>
10. **Castellanos, Argenis González. 2005.** Comparación entre Sistemas de Gestión de Bases de Datos (SGBD) bajo licenciamiento libre y comercial. 2005.
<http://www.ilustrados.com/documentos/sghbd.pdf>
11. **Ciberaula. 2006.** Introducción a Dreamweaver 3. 2006.
<http://www.programacion.net/html/tutorial/dreamweaver/2/>
12. **Corp, Microsoft. 2004.** Artículo Introducción al xml. 11 de junio de 2004.
<http://www.webexperto.com/articulos/articulo.php?cod=59>

13. **Flores, J. Tomás Nogales. 2006.** Introducción a XSLT (Extensible Stylesheet Language - Transformations). 10 de enero de 2006.
<http://www.bib.uc3m.es/~nogales/cursos/xslt.html>
14. **González., Daniel G. 2001.** GoLive frente a Dreamweaver. 1 de octubre de 2001.
<http://www.idg.es/macworld/content.asp?idart=125376>
15. **Jacobson Ivar, Booch Grady, Rumbaugh James. 2002.** El proceso unificado del software. Primera edición. Pearson Educación, S.A. . 2002.
16. **Larman, Craig.** UML y Patrones Una introducción al análisis y diseño orientado a objeto y al proceso unificado. 2da edición. Prentice Hall.
17. **Oporto, Jaime E. Villate. 2001.** Introducción al XML. 2001.
<http://quark.fe.up.pt/cursoxml/curso.pdf>
18. **Pey, Xavier. 2001.** ASP.NET. 11 de junio de 2001.
http://www.netveloper.com/imprimir.aspx?IDC=39_0
19. **Roco, Pier Angeli Zorzini. 2005.** La publicidad. febrero de 2005.
<http://www.monografias.com/trabajos20/presupuesto-publicidad/presupuesto-publicidad.shtml>
20. **Rodríguez, Toni Soto. 2002.** GaliciaCity. 11 de octubre de 2002.
<http://www.galiciacity.com/presenta.shtml>
21. **S., Christian Van Der Hest. 2001.** ¿Qué es el PHP? 23 de mayo de 2001.
<http://www.maestrosdelweb.com/editorial/phpintro/>
22. **Sanchez, María A. Mendoza. 2004.** Metodologías De Desarrollo De Software. 2004.
http://www.informatizate.net/articulos/metodologias_de_desarrollo_de_software_07062004.html

Glosario de Términos

ShockWave: Nueva tecnología de creación de banners animados, elaborada por la compañía MacroMedia.

Browser: Programa que el usuario utiliza para acceder y visualizar los servicios de Internet. Navegador.

Hit o impresión: Cada vez que una página (Banner, botón, pop up) es descargada completamente en el navegador de un usuario, se contabiliza como una impresión.

Intranet: Red propia de una organización, diseñada y desarrollada siguiendo los protocolos propios de Internet, en particular el protocolo *TCP/IP*. Puede tratarse de una red aislada, es decir no conectada a Internet.

URL (Uniform Resource Locator) -- (Localizador Uniforme de Recursos): Sistema unificado de identificación de recursos en la red. Las direcciones se componen de protocolo, FQDN y dirección local del documento dentro del servidor. Este tipo de direcciones permite identificar objetos WWW, Gopher, FTP, News,... Ejemplos de URL son: <http://cubasi.cu>.

HTML (HyperText Markup Language) -- (Lenguaje de Marcado de Hipertexto): Lenguaje en el que se escriben las páginas a las que se accede a través de navegadores WWW. Admite componentes hipertextuales y multimedia.

XML (Extensible Markup Language) --(Lenguaje Extensible de Marcado): Metalenguaje que sirve para crear lenguajes. Permite definir la gramática de lenguajes específicos (de la misma manera que HTML es a su vez un lenguaje definido por SGML). Por lo tanto XML no es realmente un lenguaje en particular, sino una manera de definir lenguajes para diferentes necesidades.

SQL (Lenguaje de Consulta Estructurado) -- (Structured Query Language): Lenguaje declarativo de acceso a bases de datos relacionales que permite especificar diversos tipos de operaciones sobre las mismas.

WSDL (Web Services Description Language): Lenguaje que escribe la forma de comunicación, es decir, los requisitos del protocolo y los formatos de los mensajes necesarios para interactuar con los servicios listados en su catálogo.

UDDI (Universal Description, Discovery, and Integration): Es uno de los estándares básicos de los

Servicios Web cuyo objetivo es ser accedido por los mensajes SOAP y dar paso a documentos WSDL.

SOAP (Simple Object Access Protocol) -- (Protocolo de Acceso a Objeto Simple): Es un protocolo para el intercambio de datos.

Tags: Es un conjunto de caracteres que se añade a un elemento de los datos para identificarlo.

FTP (File Transfer Protocol): Es un protocolo de transferencia de ficheros entre sistemas conectados a una red TCP basado en la arquitectura cliente-servidor, de manera que desde un equipo cliente nos podemos conectar a un servidor para descargar ficheros desde él o para enviarle nuestros propios archivos independientemente del sistema operativo utilizado en cada equipo.

Freeware: es un software de computadora que se distribuye sin cargo. A veces se incluye el código fuente, pero no es lo usual. Suele incluir una licencia de uso, que permite su redistribución pero con algunas restricciones.

Shareware: Una modalidad de distribución de software (juegos o programas) para que el mismo pueda ser evaluado de forma gratuita, pero generalmente por un tiempo especificado, aunque también las limitaciones pueden estar en algunas de las formas de uso o las capacidades finales.

WYSIWYG: What You See Is What You Get (en español: "lo que ves es lo que obtienes"). Para describir un editor de texto que permite escribir un documento viendo directamente el resultado final, frecuentemente el resultado impreso.

UML: Lenguaje Unificado de Modelado (UML, por sus siglas en inglés, Unified Modelling Language) es el lenguaje de modelado de sistemas de software más conocido en la actualidad.

Open source: Código libre. Software que distribuye de forma libre su código fuente, de forma que los desarrolladores pueden hacer variaciones, mejoras o reutilizarlo en otras aplicaciones. También conocido como *free software*.

GPL (General Public License): Licencia que permite el uso y modificación del código para desarrollar software libre, pero no propietario.