



UNIVERSIDAD DE LAS CIENCIAS INFORMÁTICAS
FACULTAD # 8

SISTEMA DE ANÁLISIS ESTADÍSTICO DE DATOS (SAED)
MÓDULOS ADMINISTRACIÓN Y REPORTES.



TRABAJO DE DIPLOMA PARA OPTAR POR EL TÍTULO DE INGENIERO EN CIENCIAS
INFORMÁTICAS

AUTORES

Derlys Morales Ortega.
Danisleydi Noroña Blanco.

TUTORES

Ing. Bettys L. Arias Valdés.
Ing. Aliuska Sánchez Ibarria.

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 firmo la presente a los ____ días del mes de _____ del año _____.

Firma del Autor
Danisleydi Noroña Blanco.

Firma del Autor
Derlys Morales Ortega.

Firma del Tutor
Ing. Aliuska Sánchez Ibarria.

Firma del Tutor
Ing. Bettys Leidy Arias Valdés

La UCI es una universidad modelo (...) una fuente renovadora de las posibilidades de la sociedad futura en Cuba y en el mundo (...) que tiene una responsabilidad en el desarrollo económico cubano (...) es la vanguardia intelectual y la base fundamental para el desarrollo de los recursos humanos de la informática en Cuba.

Fidel Castro Ruz.

AGRADECIMIENTOS

A todas las personas que me ayudaron, me apoyaron y estuvieron siempre a mi lado aún cuando la cima parecía estar más lejos, a aquellos que siempre creyeron en mí y no me abandonaron ni un momento, a todos ustedes dedico mi triunfo.

A mis padres, mis hermanos, mi esposa y en general a toda mi familia que me ha apoyado durante todos estos años de estudio.

A mis tutoras Aliuska y Betty por el tiempo que nos dedicaron. A Eric por su ayuda incondicional. A mi compañera de tesis.

Derlys.

Quiero agradecer primeramente a mi madre porque siempre me ha apoyado en todo y a mi padre por confiar en mí. A los dos muchas gracias por darme tanto amor y comprensión.

A mi familia, especialmente a mis hermanos Dariel y Yan marcos, a tías Esther, Virginia y Cristina, y mis primas Eleydis, Elebeydis y Gretel.

A mi novio Emilio porque siempre me ha apoyado y me enseñó que con un poco de esfuerzo todo se puede lograr.

A los vecinos que se preocuparon por mi tesis, especialmente Povea, Yaniel y Flora. A mis amistades de Bahía Honda, al Piña, Yamilé y Leyvis por estar siempre pendientes de mí. A mis amistades de la UCI, los que están y los que ya no están, que me ayudaron mucho no sólo en el desarrollo de mi tesis, sino a lo largo de estos 5 años de estudio.

A Armando por ayudarme tanto.

A los profesores de esta universidad por todo lo que me enseñaron a lo largo de mi carrera, especialmente a mis tutoras Aliuska y Betty y a la profesora Greysi por brindarnos su tiempo.

A mi compañero de tesis por compartir este tiempo de trabajo intenso.

A la Revolución y a Fidel por darme la oportunidad de estudiar en una Universidad como ésta.

Danisleydi.

DEDICATORIA

A mis padres especialmente, por haberme brindado su amor y apoyo incondicional. Por ser las personas que más admiro en el mundo, por haberme guiado siempre por el camino correcto y porque lo han dado todo por mi.

A Carlos Martínez Collado, por enseñarme a aprender.

A Eduardo Martínez Toca, por ser el mejor profesor que he tenido y por mostrarme que siempre puedo más.

Danisleydi.

A mis padres que han sabido orientar y guiar mis pasos en el transcurso de mi vida.

A mi esposa que me ha apoyado en los últimos años.

A la Revolución y a Fidel, que avizoraron la importancia económica, política y científica de la informática.

Derlys.

RESUMEN

El presente trabajo propone la realización de los módulos Administración y Reportes del Sistema de Análisis Estadístico de Datos, este es un sistema que se encarga de la gestión de los procesos de elaboración y análisis estadístico de encuestas, sin embargo este sistema no cuenta con todas las funcionalidades que debe poseer y debido a esto se procede a la elaboración de los módulos mencionados anteriormente. El módulo administrativo permite otorgar permiso de administración a determinados usuarios de los que se encuentran registrados en el sistema, permite además crear las cuentas de usuarios a partir de las solicitudes realizadas por éstos así como eliminar los elementos que lleven mucho tiempo inactivos en el sistema. Por su parte el módulo Reportes brinda a los usuarios la posibilidad de configurar los datos que desean obtener en su reporte y la fecha en la que desea que se le generen los mismos, además permite a los usuarios consultar un determinado reporte a partir de un rango de tiempo. El sistema fue desarrollado sobre la plataforma de software libre; Lenguaje de programación PHP, corriendo sobre servidor Apache y gestor de base de datos MySQL, además se utilizó la metodología RUP para la modelación.

Índice

INTRODUCCIÓN.....	1
CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA.....	4
1.1 INTRODUCCIÓN	4
1.2 ADMINISTRACIÓN DE SISTEMAS.....	4
1.3 ADMINISTRACIÓN DE USUARIOS	5
1.3.1 ADMINISTRAR LOS USUARIOS DE SITIOS WEB CON FUNCIONES.....	6
1.3.2 PERMISOS DE USUARIOS EN JOOMLA	6
1.3.3 ADMINISTRACIÓN DE ROLES	6
1.4 LOS REPORTES.....	7
1.4.1 HERRAMIENTAS PARA LA GENERACIÓN DE REPORTES: REPORTEADORES.....	7
1.5 ¿QUÉ ES PERSONALIZAR?	8
1.6 REPORTES PERSONALIZADOS	8
1.7 PLATAFORMA DE DESARROLLO	8
1.7.1 APLICACIÓN WEB.....	8
1.7.2 LA ARQUITECTURA CLIENTE / SERVIDOR.....	9
1.8 LENGUAJES DE DESARROLLO WEB.....	10
1.9 SERVIDORES WEB.....	13
1.10 GESTOR DE BASE DE DATOS.....	15
1.11 ¿POR QUÉ USAR MYSQL, PHP Y APACHE?	17
1.12 UML (UNIFIED MODELING LENGUAJE)	17
1.13 METODOLOGÍAS	18
1.14 RATIONAL ROSE	20
1.15 CONCLUSIONES.....	20
CAPÍTULO 2: CARACTERÍSTICAS DEL SISTEMA.....	21
2.1 INTRODUCCIÓN	21
2.2 FLUJO ACTUAL DE LOS PROCESOS.....	21
2.3 OBJETO DE AUTOMATIZACIÓN.....	21
2.4 PROPUESTA DE SISTEMA.....	22
2.5 MODELO DE NEGOCIO.....	22
2.5.1 DESCRIPCIÓN GENERAL DE LOS PROCESOS DE NEGOCIO.....	22
2.5.2 REGLAS DEL NEGOCIO	22
2.5.3 REPRESENTACIÓN DE LOS CASOS DE USO DEL NEGOCIO.....	23
2.5.3.1 Descripción textual de los casos de uso del negocio.....	23
2.5.3.2 Diagrama de Casos de Uso del Negocio.....	25
2.5.3.3 Diagramas de actividades.....	26

2.5.4 DIAGRAMA DE CLASES DEL MODELO DE OBJETOS	28
2.6 ESPECIFICACIÓN DE LOS REQUISITOS DE SOFTWARE.....	29
2.6.1 REQUERIMIENTOS FUNCIONALES	29
2.6.2 REQUERIMIENTOS NO FUNCIONALES.	30
2.6.3 DEFINICIÓN DE LOS CASOS DE USO.	32
2.6.3.1 <i>Definición de los actores del sistema</i>	32
2.6.3.2 <i>Descripción textual de los casos de uso</i>	32
2.6.3.3 <i>Diagrama de casos de uso</i>	40
2.7 CONCLUSIONES	41
CAPÍTULO 3: ANÁLISIS Y DISEÑO DEL SISTEMA.....	42
3.1 INTRODUCCIÓN	42
3.2 ANÁLISIS.....	42
3.2.1 MODELO DE CLASES DE ANÁLISIS.	42
3.2.2 DIAGRAMAS DE INTERACCIÓN.....	46
3.3 DISEÑO.....	46
3.3.1 DISEÑO ARQUITECTÓNICO	46
3.3.2 DIAGRAMAS DE INTERACCIÓN.....	47
3.3.3 DIAGRAMAS DE CLASES.....	47
3.3.4 DESCRIPCIÓN DE LAS CLASES.	58
3.3.5 DISEÑO DE LA BD	64
3.3.5.1 <i>Descripción de las tablas</i>	65
3.3.6 PRINCIPIOS DE DISEÑO.	67
3.3.7 TRATAMIENTO DE ERRORES	68
3.3.8 SEGURIDAD.....	68
3.3.9 CONCEPCIÓN DE LA AYUDA.	69
3.4 CONCLUSIONES	69
CAPÍTULO 4: IMPLEMENTACIÓN Y PRUEBAS.	70
4.1 INTRODUCCIÓN	70
4.2 IMPLEMENTACIÓN.....	70
4.2.1 DIAGRAMA DE DESPLIEGUE.	70
4.2.2 ESTRUCTURA DEL MODELO DE IMPLEMENTACIÓN	71
4.2.3 DIAGRAMAS DE COMPONENTES	72
4.3 MODELO DE PRUEBA.....	74
4.3.1 PRUEBAS DE CAJA NEGRA	74
4.3.2 CASOS DE PRUEBA.....	75
4.4 CONCLUSIONES	78
CONCLUSIONES.....	79
RECOMENDACIONES.....	80
REFERENCIAS BIBLIOGRÁFICAS.....	81
BIBLIOGRAFÍA.....	83

GLOSARIO DE TÉRMINOS	84
ANEXOS	87
ANEXO # 1 DIAGRAMAS DE COLABORACIÓN DEL ANÁLISIS	87
ANEXO # 2 DIAGRAMAS DE SECUENCIAS DEL DISEÑO	92

Índice de Tablas

TABLA 2. 1 ACTORES DEL NEGOCIO.....	23
TABLA 2. 2 TRABAJADORES DEL NEGOCIO.....	23
TABLA 2. 3 DESCRIPCIÓN TEXTUAL CU NEGOCIO: OBTENER REPORTE.....	23
TABLA 2. 4 DESCRIPCIÓN TEXTUAL CU NEGOCIO: CONSULTAR REPORTE.....	24
TABLA 2. 5 ACTORES DEL SISTEMA.....	32
TABLA 2. 6 DESCRIPCIÓN TEXTUAL CU SISTEMA: SOLICITAR CUENTA.....	32
TABLA 2. 7 DESCRIPCIÓN TEXTUAL CU SISTEMA: AUTENTICAR USUARIO.....	33
TABLA 2. 8 DESCRIPCIÓN TEXTUAL CU SISTEMA: CONFIGURAR REPORTE.....	34
TABLA 2. 9 DESCRIPCIÓN TEXTUAL CU SISTEMA: CONSULTAR REPORTE.....	34
TABLA 2. 10 DESCRIPCIÓN TEXTUAL CU SISTEMA: ELIMINAR ELEMENTOS INNECESARIOS.....	35
TABLA 2. 11 DESCRIPCIÓN TEXTUAL CU SISTEMA: ENVIAR REPORTE.....	36
TABLA 2. 12 DESCRIPCIÓN TEXTUAL CU SISTEMA: ELIMINAR ELEMENTOS INACTIVOS.....	36
TABLA 2. 13 DESCRIPCIÓN TEXTUAL CU SISTEMA: CREAR CUENTA.....	38
TABLA 2. 14 DESCRIPCIÓN TEXTUAL CU SISTEMA: EDITAR PERFIL.....	38
TABLA 2. 15 DESCRIPCIÓN TEXTUAL CU SISTEMA: MOSTRAR USUARIOS REGISTRADOS.....	39
TABLA 2. 16 DESCRIPCIÓN TEXTUAL CU SISTEMA: ASIGNAR ADMINISTRACIÓN.....	39
TABLA 3. 1 CLASE CONT_USUARIO.....	58
TABLA 3. 2 CLASE CONT_REPORTER.....	59
TABLA 3. 3 CLASE CONT_ENCUESTAS.....	60
TABLA 3. 4 CLASE CONEXION.....	61
TABLA 3. 5 CLASE CENCUESTA.....	61
TABLA 3. 6 CLASE CUSUARIO.....	62
TABLA 3. 7 CLASE CREPORTE.....	62
TABLA 3. 8 CLASES INTERFAZ DE USUARIO.....	63
TABLA 3. 9 REPORTE.....	65
TABLA 3. 10 ESTADÍGRAFO.....	66
TABLA 3. 11 TIPO _ ESTADÍGRAFO.....	66
TABLA 3. 12 ESTADIGRAFOPREGUNTA.....	66
TABLA 3. 13 REPORTEESTADIGRAFO.....	66
TABLA 3. 14 REPORTE_MOSTRAR.....	66
TABLA 3. 15 ADMINISTRADORES.....	66
TABLA 3. 16 CUENTA.....	66
TABLA 3. 17 SOLICITUD_CUENTA.....	67
TABLA 4. 1 CASO DE PRUEBA CU: AUTENTICAR USUARIO.....	75
TABLA 4. 2 CASO DE PRUEBA CU: SOLICITAR CUENTA.....	76
TABLA 4. 3 CASO DE PRUEBA CU: EDITAR PERFIL.....	77

ÍNDICE DE FIGURAS

FIGURA 2. 1 DIAGRAMA DE CASOS DE USO DEL NEGOCIO.....	26
FIGURA 2. 2 DIAGRAMA DE ACTIVIDADES CU: OBTENER REPORTE.....	27
FIGURA 2. 3 DIAGRAMA DE ACTIVIDADES CU: CONSULTAR REPORTE.....	28
FIGURA 2. 4 DIAGRAMA DE CLASES DEL MODELO DE OBJETOS.....	29
FIGURA 2. 5 DIAGRAMA DE CASOS DE USO DEL SISTEMA.....	41
FIGURA 3. 1 DIAGRAMA DE CLASES DEL ANÁLISIS CU: AUTENTICAR USUARIO.....	42
FIGURA 3. 2 DIAGRAMA DE CLASES DEL ANÁLISIS CU: CONFIGURAR REPORTE.....	43
FIGURA 3. 3 DIAGRAMA DE CLASES DEL ANÁLISIS CU: CONSULTAR REPORTE.....	43
FIGURA 3. 4 DIAGRAMA DE CLASES DEL ANÁLISIS CU: ENVIAR REPORTE.....	43
FIGURA 3. 5 DIAGRAMA DE CLASES DEL ANÁLISIS CU: SOLICITAR CUENTA.....	44
FIGURA 3. 6 DIAGRAMA DE CLASES DEL ANÁLISIS CU: CREAR CUENTA.....	44
FIGURA 3. 7 DIAGRAMA DE CLASES DEL ANÁLISIS CU: ELIMINAR ELEMENTOS INACTIVOS.....	44
FIGURA 3. 8 DIAGRAMA DE CLASES DEL ANÁLISIS CU: EDITAR PERFIL.....	45
FIGURA 3. 9 DIAGRAMA DE CLASES DEL ANÁLISIS CU: ELIMINAR ELEMENTOS INNECESARIOS.....	45
FIGURA 3. 10 DIAGRAMA DE CLASES DEL ANÁLISIS CU: MOSTRAR USUARIOS REGISTRADOS.....	45
FIGURA 3. 11 DIAGRAMA DE CLASES DEL ANÁLISIS CU: ASIGNAR ADMINISTRACIÓN.....	46
FIGURA 3. 12 DIAGRAMA DE CLASES DEL DISEÑO CU: AUTENTICAR USUARIO.....	48
FIGURA 3. 13 DIAGRAMA DE CLASES DEL DISEÑO CU: CREAR CUENTA.....	49
FIGURA 3. 14 DIAGRAMA DE CLASES DEL DISEÑO CU: EDITAR PERFIL.....	50
FIGURA 3. 15 DIAGRAMA DE CLASES DEL DISEÑO CU: ASIGNAR ADMINISTRACIÓN.....	51
FIGURA 3. 16 DIAGRAMA DE CLASES DEL DISEÑO CU: ELIMINAR ELEMENTOS INACTIVOS.....	52
FIGURA 3. 17 DIAGRAMA DE CLASES DEL DISEÑO CU: ELIMINAR ELEMENTOS INNECESARIOS.....	53
FIGURA 3. 18 DIAGRAMA DE CLASES DEL DISEÑO CU: MOSTRAR USUARIOS REGISTRADOS.....	54
FIGURA 3. 19 DIAGRAMA DE CLASES DEL DISEÑO CU: SOLICITAR CUENTA.....	55
FIGURA 3. 20 DIAGRAMA DE CLASES DEL DISEÑO CU: CONFIGURAR REPORTE.....	56
FIGURA 3. 21 DIAGRAMA DE CLASES DEL DISEÑO CU: ENVIAR REPORTE.....	57
FIGURA 3. 22 DIAGRAMA DE CLASES DEL DISEÑO CU: CONSULTAR REPORTE.....	58
FIGURA 3. 23 DIAGRAMA ENTIDAD-RELACIÓN.....	65
FIGURA 4. 1 DIAGRAMA DE DESPLIEGUE.....	70
FIGURA 4. 2 ESTRUCTURA DEL MODELO DE IMPLEMENTACIÓN.....	71
FIGURA 4. 3 DIAGRAMA DE COMPONENTES: GESTIONAR CUENTA.....	72
FIGURA 4. 4 DIAGRAMA DE COMPONENTES: GESTIONAR REPORTE.....	73
FIGURA 4. 5 DIAGRAMA DE COMPONENTES: GESTIONAR ENCUESTA.....	74

INTRODUCCIÓN

En la Facultad 8 se encuentra implementado un sistema para el análisis estadístico de datos, específicamente de encuestas online, conocido como SAED. Este sistema es capaz de gestionar los procesos de elaboración y análisis estadístico de encuestas. Sin embargo no dispone de un módulo administrativo capaz de eliminar los usuarios que no se hayan registrado en el mismo por mucho tiempo o las encuestas que no hayan sido publicadas en un determinado período, lo que provoca que la base de datos esté cargada con información que no se encuentre actualizada y ésto a su vez trae consigo que el sistema pierda en eficiencia.

SAED cuenta solamente con usuarios locales lo que provoca cierto grado de incomodidad debido a la necesidad de que cada usuario tenga que introducir todos sus datos para poder registrarse y que además tenga que memorizar dos contraseñas, la que posee para autenticarse en el dominio UCI y con la que se autentica en el SAED. El hecho de que cada usuario tenga que introducir sus datos para registrarse puede provocar la existencia de errores en dicha información y estos datos erróneos se incorporan al sistema.

Además el sistema brinda un análisis de los resultados de cada una de las encuestas, pero no existe la posibilidad de que el usuario reciba un reporte, en los horarios solicitados por él, que contenga los parámetros que éste necesite de todo el análisis, o sea que el usuario no puede obtener el estado en que se encuentra su encuesta en el momento que desee. La ausencia de estos reportes puede provocar inconformidad por parte de los encuestadores debido a que éstos no ven satisfechas sus necesidades reales en el momento preciso o sea no poseen una vía para obtener la información que necesiten en el momento que la deseen.

El SAED es un sistema muy útil para los usuarios que deseen publicar encuestas y obtener un análisis estadístico de las mismas sin embargo es evidente que aún no cuenta con todas las funcionalidades que lo hagan trabajar de forma óptima y que permitan a los encuestadores satisfacer todas sus exigencias con respecto a las encuestas que publiquen.

Para solucionar los problemas existentes surge este trabajo proponiendo cómo gestionar los procesos de administración y envío de reportes del Sistema de Análisis Estadístico de Datos (SAED). Para ello se defiende que si se implementan los módulos administración y reporte del Sistema de Análisis Estadístico de Datos, garantizándoles seguridad a los usuarios e información del estado de cada encuesta entonces será posible obtener una mayor seguridad, fiabilidad y eficiencia de dicho sistema.

Con el desarrollo del presente trabajo se espera obtener una mayor velocidad en el procesamiento de datos del sistema, al ser eliminados los usuarios que no se hayan registrado frecuentemente y/o las encuestas no publicadas en un determinado período así como la posibilidad de brindar información de las encuestas a los usuarios mediante el envío de reportes lo que traería consigo una mayor conformidad de los usuarios con el SAED al obtener éstos los reportes con la información que realmente necesitan en el momento preciso. Por otro lado se espera obtener además toda la documentación teórica que sea necesaria.

Para ello ha sido necesario realizar un estudio de los procesos de administración de sistemas y la gestión de reportes y más específicamente los procesos de administración de usuarios y la obtención de reportes personalizados.

El objetivo general que se persigue con el presente trabajo es realizar el análisis, diseño e implementación de los módulos administración y reporte del Sistema de Análisis Estadístico de Datos (SAED) y para el cumplimiento de dicho objetivo se plantean como objetivos específicos:

1. Realizar la búsqueda y el estudio de la información referente a los procesos de administración de sistemas y envío de reportes.
2. Realizar un estudio del Sistema de Análisis Estadístico de Datos (SAED).
3. Realizar el análisis, diseño e implementación de los módulos en proposición.

De esta misma forma se definieron un grupo de tareas de investigación que tributan al cumplimiento de cada uno de los objetivos específicos anteriores, éstas son:

1. Realizar una búsqueda de la información necesaria para la administración y envío de reportes.
2. Estudiar minuciosamente los procesos de administración de sistemas.
3. Realizar un estudio de las herramientas necesarias para el envío de reportes.
4. Realizar un estudio del estado del arte teniendo en cuenta el trabajo ya realizado sobre el Sistema de Análisis Estadístico de Datos (SAED).
5. Realizar un estudio de las principales tendencias y tecnologías actuales a utilizar.
6. Estudiar la metodología RUP ya utilizada en la versión anterior y otras que sean necesarias para posterior valoración como proceso de desarrollo del software así como los posibles lenguajes y herramientas a utilizar.
7. Realizar el modelo de Negocios y el levantamiento de Requisitos.
8. Realizar el análisis y el diseño de los módulos Administración y Reporte.

9. Estructurar la investigación en el formato adecuado para obtener una buena documentación.
10. Implementar los módulos para desarrollar más funcionalidades en la plataforma.
11. Realizar pruebas al sistema.

El presente trabajo está compuesto por cuatro capítulos:

- Capítulo 1 - Fundamentación teórica: incluye el estado del arte del tema tratado, además se describe el objeto de estudio, el campo de acción y la situación problemática presente, así como la selección de la plataforma, lenguaje, herramientas y metodología con las que se llevará a cabo la propuesta.
- Capítulo 2 - Características del sistema: se modela el negocio y se hace el levantamiento de requisitos concluyendo con el diagrama de casos de uso del sistema.
- Capítulo 3 - Análisis y diseño del sistema: se determinan y describen las clases y se realiza el diseño de la bases de datos.
- Capítulo 4 – Implementación y prueba: se hace referencia al modelo de implementación mediante el diagrama de despliegue y de componentes; se describen los casos de prueba de caja negra, encaminadas a probar las funcionalidades del software.

Capítulo 1: Fundamentación Teórica.

1.1 Introducción

En este capítulo se realiza un estudio de la administración de sistemas, así como de la obtención de reportes a partir de varias herramientas reporteadoras y a su vez se estudian las tendencias, técnicas, tecnologías, metodologías y software usados en la actualidad y se profundiza en las que se han utilizado para la solución del problema.

1.2 Administración de sistemas

La administración de sistema abarca un gran número de responsabilidades y éstas varían enormemente de una organización a otra pero, por lo general a los administradores se les encomienda la instalación, soporte y mantenimiento de los servidores u otros sistemas de cómputo, la planeación de respuesta a contingencias y otros problemas. Además se incluyen en ocasiones responsabilidades como la programación, el manejo de proyectos relacionados con el sistema, la supervisión o entrenamiento de operadores de cómputo y ser el consultor para los problemas que se encuentran más allá del conocimiento técnico del personal de soporte. Debido a este gran número de responsabilidades la persona encargada de la administración de sistemas debe poseer tanto habilidades técnicas como responsabilidad.

1.2.1 Filosofía de la administración de sistemas

Aún cuando los detalles específicos de la administración de sistemas pueden variar entre plataformas, hay temas subyacentes que no. Estos temas conforman la filosofía de la administración de sistemas.(Productos)

Los temas son:

Automatizar todo: En la gran mayoría de los casos los sistemas y los usuarios superan el número de administradores de sistemas y la mejor forma que tienen éstos de mantenerse actualizados es la automatización. Por lo general cualquier cosa que se realice más de una vez debe ser examinada como candidata a automatización.

Documentar todo: Es muy importante documentar todo ya que con el tiempo se olvida lo que debía estar documentado y hay una mayor pérdida de tiempo tratando de reaprender lo que se ha olvidado o reparando lo que ha salido mal debido a un entendimiento incompleto de la situación.

Comunicar tanto como sea posible: Los cambios que para un administrador de sistemas pueden parecer insignificantes no lo son para sus usuarios y es por eso que es importante buscar la forma para informar a los usuarios sobre lo que se va a hacer, lo que se está haciendo y lo que se ha hecho.

Conocer sus recursos: La administración de sistemas es mayormente un asunto de balancear los recursos disponibles con la gente y los programas que utilizan esos recursos por lo que es muy importante conocer completamente los recursos existentes.

Conocer sus usuarios: Los usuarios son las personas que utilizan los sistemas y recursos sobre los que un administrador tiene responsabilidad y constituyen la clave en la habilidad de administrar exitosamente por lo que sin entender a los usuarios, no se pueden entender los recursos que ellos requieren.

Conocer el negocio: El punto clave aquí es entender el propósito de los sistemas que se están administrando en un sentido global o sea las aplicaciones que se deben ejecutar en un período de tiempo particular, los tiempos durante los que se ha efectuado mantenimientos al sistema o las nuevas tecnologías que se podrían utilizar para resolver viejos problemas de negocios.

La seguridad no puede ser una ocurrencia posterior: La seguridad no se puede asumir como algo garantizado ya que hasta los sistemas independientes que no están conectados a la Internet están a riesgo. Por lo tanto, es extremadamente importante considerar las implicaciones de seguridad en todo lo que se realice.

Planifique: Es imposible predecir el futuro con un 100% de fidelidad sin embargo con la planificación puede estar preparado para los cambios que se realicen.

Espera lo inesperado: Habrá ocasiones en las que el administrador de sistema puede ser tomado por sorpresa, la mejor solución para esta dificultad es la flexibilidad o sea hacer el trabajo de forma tal que brinde la mayor cantidad de opciones tanto al administrador como a los usuarios.

1.3 Administración de usuarios

La administración de usuarios en cualquier entorno es una vía muy eficaz no sólo para mantener la seguridad del sistema sino también para lograr la confiabilidad por parte de los usuarios y un buen rendimiento de la aplicación.

Existen diferentes formas de implementar la administración de los usuarios y todas brindan, de una forma u otra, la posibilidad de mantener un control sobre los privilegios necesarios para cada usuario de una aplicación, evitando de esta forma que las personas tengan acceso a funcionalidades que no le son necesarias para desarrollar las actividades que requieran del sistema.

A continuación se muestran varias de las formas para implementar la administración de usuarios.

1.3.1 Administrar los usuarios de sitios Web con funciones

Esta administración se aplica mayormente en aplicaciones ASP.NET que requieren de suscripción. El establecimiento de funciones es muy útil cuando la aplicación admite un gran número de usuarios o cuando es probable que la lista de usuarios cambie con el tiempo.

En el contexto de administración de sistemas una función viene siendo el nombre de un grupo como por ejemplo administradores o recepcionistas. Una vez que se hayan establecido estas funciones se le asigna usuarios a cada función y se le conceden permisos a éstas y los usuarios que pertenezcan a dicha función heredan los permisos de la misma.

Es por esto que se plantea que las funciones son una manera eficaz de administrar los permisos de los grupos de usuarios.

1.3.2 Permisos de usuarios en Joomla

Joomla tiene dos jerarquías principales de grupos de usuarios uno para acceder a las secciones designadas del sitio Web y uno para acceder a la administración de Joomla. Estos grupos se organizan de forma jerárquica o piramidal de modo que los permisos que le corresponden a un tipo de usuario le pertenecen también a los usuarios de rango superior.

1.3.3 Administración de roles

En este tipo de administración un rol se compone de un número de tareas, responsabilidades o conocimientos necesarios para realizar un trabajo específico. En este caso se convierte en una lista de uno o más pares de objetos y uno o más permisos de acceso que se aplican al objeto. Por ejemplo (administrador: permisos 1, 2, 3, 4). Para que un rol se pueda utilizar, debe estar activado. Mediante esta forma de administración se puede lograr que la seguridad de un sistema cumpla de forma óptima los requisitos de seguridad en cuanto a escalabilidad, simplicidad y versatilidad.

1.4 Los reportes

Un reporte, en general, se define como un informe que se emite o se presenta por una persona, entidad o sistema. Crear reportes es un proceso crítico para tareas y proyectos de investigación.

El reporte de investigación es un documento donde se presenta el resultado de un estudio en torno a un tema específico. Dicho estudio pretende dar respuesta a una(s) pregunta(s) y con ello ampliar el conocimiento respecto a dicho tema. El reporte contiene información recabada de diferentes fuentes de información a través de diversos medios como pueden ser: entrevistas, encuestas, observaciones, medios electrónicos, entre otros. Dicha información aparece organizada, relacionada de una manera que deja ver aspectos nuevos o distintos sobre la temática abordada. (Ruvalcaba).

1.4.1 Herramientas para la generación de reportes: Reporteadores.

Crear reportes es crítico para tareas y proyectos de investigación, es por eso que existen varias herramientas empleadas para la obtención de los mismos en diferentes plataformas, algunas de las más utilizadas son:

Crystal Reports 10

Es la herramienta de elaboración de informes estándar para Visual Studio .NET. Esta herramienta permite crear contenido interactivo con calidad de presentación en la plataforma .NET. Con la utilización de Crystal Reports 10 se pueden almacenar informes en plataformas Windows y Web, así como publicar informes de Crystal como servicios Web de informes en un servidor Web. Por ejemplo, se puede crear una aplicación Web que permita a los usuarios profundizar en un gráfico y filtrar la información en función de sus necesidades, en este caso el gráfico es un informe de Crystal que interactúa con otros controles de la aplicación.

Jasper Report:

Es una librería para la generación de informes. Está escrita en java y es libre. El funcionamiento consiste en escribir un xml donde se recogen las particularidades del informe. Este xml lo tratan las clases del Jasper para obtener una salida. Esta salida puede ser un PDF, XML, HTML, CSV, XLS, RTF, TXT. Otra ventaja de utilizar Jasper Report es que se integra perfectamente con el JFreeChart que es una librería libre para la generación de todo tipo de gráficas.

Rpv:

Esta herramienta es utilizada para crear reportes en cualquier lenguaje de programación y en cualquier sistema operativo. Con ésta se pueden generar facturas que incluyen códigos de barras, imágenes, gráficos o enlaces a Internet o a otros reportes. Además Rpv Business Reports incluye muchas herramientas externas que ayudan a administrar los reportes.

PHP Report Maker:

Es una poderosa herramienta reporteadora que puede generar reportes Web en PHP dinámicamente desde bases de datos MySQL. Además se pueden crear reportes resumidos y detallados en tiempo real o reportes cruzados de páginas Web. También soporta graficación básica en Flash (Columnas, barras, líneas o pay). Las páginas Web son generadas como PHP puro por lo que no se requieren componentes de servidor o de cliente.

1.5 ¿Qué es personalizar?

Existen varias definiciones para este término en la Web, por ejemplo el Diccionario de Marketing Directo define la personalización como la interpelación directa al receptor de la publicidad directa por su nombre u otros datos personales. La personalización garantiza a su oferta un alto nivel de atención.(Directo 1999-2008)

Sin embargo la definición que está más acorde con el tema de desarrollo del presente trabajo es la que nos indica que personalizar no es más que configurar todas las opciones que ofrece una aplicación según las preferencias del usuario.(Observatorio 2008) . O sea mediante la personalización un sistema va a ser capaz de responder a las necesidades de cada usuario en particular.

1.6 Reportes personalizados

Después de conocer lo que es personalizar se puede llegar fácilmente a la definición de reporte personalizado, éste no es más que un reporte que se obtiene a partir de las necesidades y exigencias de cada usuario o sea el reporte va a estar constituido por los aspectos que sean de interés para el usuario.

1.7 Plataforma de desarrollo

1.7.1 Aplicación Web.

Las aplicaciones Web son aquellas que los usuarios usan accediendo a un servidor Web a través de Internet o de una intranet. Las aplicaciones Web son populares debido a la practicidad del navegador

Web como cliente ligero. La habilidad para actualizar y mantener aplicaciones Web sin distribuir e instalar software en miles de potenciales clientes es otra razón de su popularidad (PROAÑO 2006).

Una aplicación Web es un sistema donde la entrada o acción del usuario afecta el estado del negocio. Su arquitectura general es la de un sistema cliente/servidor, donde tanto el cliente (el navegador) como el servidor (el servidor Web), y el protocolo mediante el que se comunican (el HTTP) son estándar, y no han de ser creados por el desarrollador. Una de sus ventajas más significativas es su forma de instalación y distribución. Normalmente instalar una aplicación de este tipo consiste en configurar los componentes del lado del servidor en la red y no es necesaria una instalación o configuración en el lado cliente (Barroso 2006).

Ventajas de las aplicaciones Web (PROAÑO 2006).

- Aportan interoperabilidad entre aplicaciones de software de sus propiedades o de las plataformas sobre las que se instalen.
- Los servicios Web fomentan los estándares y protocolos basados en texto, que hacen más fácil acceder a su contenido y entender su funcionamiento.
- Al apoyarse en HTTP, los servicios Web pueden aprovecharse de los sistemas de seguridad firewall sin necesidad de cambiar las reglas de filtrado.
- Permiten que servicios y software de diferentes compañías ubicadas en diferentes lugares geográficos puedan ser combinados fácilmente para proveer servicios integrados.
- Permiten la interoperabilidad entre plataformas de distintos fabricantes por medio de protocolos estándar.

1.7.2 La arquitectura cliente / servidor.

La arquitectura cliente/servidor es un modelo para el desarrollo de sistemas de información, en el que las transacciones se dividen en procesos independientes que cooperan entre sí para intercambiar información, servicios o recursos. Se denomina cliente al proceso que inicia el diálogo o solicita los recursos y servidor, al proceso que responde a las solicitudes.

Siempre que un cliente requiere un servicio lo solicita al servidor correspondiente y éste, le responde proporcionándolo. Normalmente, pero no necesariamente, el cliente y el servidor están ubicados en

distintos procesadores. Los clientes se suelen situar en ordenadores personales o estaciones de trabajo y los servidores en procesadores departamentales o de grupo.

La mayoría de los sistemas Cliente/Servidor actuales, se basan en redes locales y por lo tanto utilizan protocolos no orientados a conexión, lo cual implica que las aplicaciones deben hacer las verificaciones. La red debe tener características adecuadas de desempeño, confiabilidad, transparencia y administración (Lascano 2007).

Entre las principales características de la arquitectura cliente / servidor, se pueden destacar las siguientes:

- El servidor presenta a todos sus clientes una interfaz única y bien definida.
- El cliente no necesita conocer la lógica del servidor, sólo su interfaz externa.
- El cliente no depende de la ubicación física del servidor, ni del tipo de equipo físico en el que se encuentra, ni de su sistema operativo.
- Los cambios en el servidor implican pocos o ningún cambio en el cliente.

Ventajas de la arquitectura cliente/servidor.

- El servidor no necesita potencia de procesamiento, parte del proceso se reparte con los clientes.
- Se reduce el tráfico de red considerablemente. Idealmente, el cliente se conecta al servidor cuando es estrictamente necesario, obtiene los datos que necesita y cierra la conexión dejando la red libre.

Por todas las razones antes expuestas, el sistema se continuará implementando como una aplicación Web.

1.8 Lenguajes de desarrollo Web

Las aplicaciones Web, producto del desarrollo vertiginoso que han sufrido, poseen innumerables herramientas para su desarrollo (Martin Enrique 2002). A continuación se presentan varios lenguajes de programación que se utilizan para la construcción de sitios Web.

PHP

PHP (acrónimo de "PHP: Hypertext Preprocessor") es un lenguaje interpretado de alto nivel ejecutado en el servidor. Lo mejor de usar PHP es que es extremadamente simple y ofrece muchas características avanzadas para los programadores profesionales.

PHP fue creado por Rasmus Lerdorf a finales de 1994, aunque no hubo una versión utilizable por otros usuarios hasta principios de 1995. Esta primera versión se llamó, Personal Home Page Tools.

Dispone de múltiples herramientas que permiten acceder a bases de datos de forma sencilla, por lo que es ideal para crear aplicaciones para Internet.

PHP es un lenguaje "open source" y 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 facilidad de funcionar tanto para Unix (con Apache) como para Windows (con Microsoft Internet Information Server) de forma que el código que se haya creado para una de ellas no tiene por qué modificarse al pasar a la otra.

PHP soporta la mayoría de servidores Web de hoy en día, incluyendo Apache, Microsoft Internet Information Server, Personal Web Server, Netscape y iPlanet, Oreilly Website Pro server, Caudium, Xitami, OmniHTTPd y muchos otros.

PHP también brinda la posibilidad de usar programación de procedimientos ó programación orientada a objetos.

Entre las habilidades de PHP se incluyen, creación de imágenes, ficheros PDF y películas Flash (usando libswf y Ming). También se pueden presentar otros resultados, como XHTML y ficheros XML. PHP puede autogenerar estos ficheros y grabarlos en el sistema de ficheros en vez de presentarlos en la pantalla.

El resultado es normalmente una página HTML. Por lo que al usuario le parecerá que está visitando una página HTML que cualquier navegador puede interpretar.

Al ser PHP un lenguaje que se ejecuta en el servidor no es necesario que el navegador lo soporte, es independiente del navegador, pero sin embargo para que sus páginas PHP funcionen, el servidor donde están alojadas debe soportar PHP (Saether Bakken 2003).

JSP

JSP es un acrónimo de Java Server Pages, es una tecnología orientada a crear páginas Web con programación en Java.

Con JSP podemos crear aplicaciones Web que se ejecuten en variados servidores Web, de múltiples plataformas, ya que Java es en esencia un lenguaje multiplataforma. Las páginas JSP están compuestas de código HTML/XML mezclado con etiquetas especiales para programar scripts en sintaxis Java.(Felipe 02-2004).

En general, JSP permite a los desarrolladores distribuir de una manera fácil la funcionalidad de la aplicación a una gran gama de programadores. Estos programadores no tienen por qué tener conocimientos del lenguaje de programación Java o del código servlet, pudiendo de esta manera concentrarse tan sólo en HTML mientras se crean las aplicaciones lógicas.

PERL

Es un lenguaje de programación muy utilizado para construir aplicaciones CGI para el Web. Perl es un acrónimo de Practical Extracting and Reporting Language, indicando que se trata de un lenguaje de programación muy práctico para extraer información de archivos de texto y generar informes a partir del contenido de los ficheros(2001).

Es un lenguaje libre de uso. Antes estaba muy asociado a la plataforma Unix, pero en la actualidad está disponible en otros sistemas operativos como Windows. Perl es un lenguaje de programación interpretado, al igual que muchos otros lenguajes de Internet como JavaScript o como ASP.

ASP

ASP (Active Server Pages) es la tecnología desarrollada por Microsoft para la creación de páginas dinámicas del servidor. ASP se escribe en la misma página Web, utilizando el lenguaje Visual Basic Script o Jscript (de Microsoft).

La mayor desventaja que presenta este lenguaje es que sólo se puede implementar en los Servidores Web de su desarrollador: Microsoft. La segunda versión de ASP: el ASP.NET, comprende algunas mejoras en cuanto a posibilidades del lenguaje y rapidez. ASP.NET tiene algunas diferencias en cuanto a sintaxis con el ASP. Para implementarlo es necesario montar en el Servidor la Plataforma .NET(2001).

Lenguaje seleccionado

Luego de hacer el análisis entre el PHP y el ASP, se decide utilizar el PHP puesto que:

Para ejecutar ASP se necesita el Internet Information Server instalado en servers con plataforma Windows, el cual como todos conocemos no es libre. El PHP puede correr sobre Windows pero también funciona en Linux, Solaris. En cuanto a la conectividad a bases de datos, PHP normalmente usa MySQL mientras que el ASP utiliza normalmente el SQL Server producto propietario. Podemos citar que el PHP es mucho más rápido ya que ASP basa su arquitectura en tecnología COM que sobrecarga el funcionamiento del servidor donde se encuentra.

El PHP no tiene costo alguno, o sea que cuando se adquiere incluye un sinnúmero de bibliotecas que proporcionan el soporte para la mayoría de las aplicaciones Web, por ejemplo e-mail, generación de ficheros PDF y otros. En caso de que no se tengan las bibliotecas éstas se pueden encontrar gratis en Internet. En el caso de ASP forma parte del Internet Information Server que viene integrado en el sistema operativo Windows, lo cual implica un elevado costo de adquisición.

PHP está basado en el lenguaje C++ y la sintaxis usada es muy similar a C/C++ el cual es considerado aún el mejor lenguaje de programación por muchos programadores. Por otra parte, el ASP es basado en el Visual Basic que solo Microsoft produce.

PHP y ASP son parecidos en cuanto a la forma de utilización, pero PHP es más rápido, gratuito y multiplataforma.

1.9 Servidores Web

Un servidor Web es un software que suministra páginas Web en respuesta a las peticiones de los navegadores Web.

IIS

Internet Information Services/Server (IIS): son servicios para los ordenadores que funcionan con Windows. Originalmente era parte del Option Pack para Windows NT. Luego fue integrado en otros sistemas operativos de Microsoft destinados a ofrecer servicios, como Windows 2000 o Windows Server 2003. Windows XP Profesional incluye una versión limitada de IIS. Ofrece servicios como FTP, SMTP, NNTP y HTTP/HTTPS. Procesa páginas de ASP y ASP.NET; aunque también puede incluir PHP o Perl.(Manso 2000) Una de las desventajas que tiene es que solamente se puede utilizar en sistemas de Windows.

Lighttpd

Lighttpd es un servidor Web diseñado de forma monolítica, y no emplea hilos para atender las distintas peticiones. Lo que afirman sus desarrolladores es que tiene un consumo de memoria de un quinto respecto Apache y que las páginas estáticas las sirve más rápido (de 4 a 6 veces). (2007)

Sus funcionalidades se habilitan como módulos en un único fichero de configuración, muy pequeño. Cuenta con módulos de Apache como el mod_userdir, el mod_access, el mod_proxy o el mod_rewrite aunque hay algunas funcionalidades que presenta Apache y que Lighttpd aún no tiene desarrolladas como por ejemplo mod_webdav.

Lighttpd soporta PHP sin embargo no lo hace de una forma muy sencilla.

Apache

Apache es un servidor web que, por su configurabilidad, robustez y estabilidad hace que cada vez millones de servidores reiteren su confianza en él.

Dentro de sus puntos fuertes se encuentran:

- Tiene interfaz de autenticación con todos los sistemas.
- Facilita la integración como "plug-ins" de lenguajes de programación de páginas web dinámicas.
- Tiene integración en estándar del protocolo de seguridad SSL.
- Provee interfaz a todas las bases de datos.
- Servidor altamente configurable de diseño modular: se pueden escribir módulos para realizar determinadas funciones lo que implica que haya gran cantidad de ellos disponibles para su utilización.

El Apache fue hecho para proveer un alto grado de calidad y fortaleza para las implementaciones que utilizan el protocolo HTTP. Está ligado a la plataforma (Linux, Windows, UNIX) sobre la cual los individuos o instituciones pueden construir sistemas confiables con fines experimentales o para resolver un problema específico de la organización (Kabir 2003).

1.10 Gestor de base de datos

Un Sistema de Gestión de Base de Datos(SGBD) es el software que permite la utilización y actualización de los datos almacenados en una (o varias) base(s) de datos por uno o varios usuarios desde diferentes puntos de vista y a la vez.

Microsoft SQL Server

Microsoft SQL Server, propietario de Microsoft, pertenece a la familia de los sistemas de administración de base de datos, operando en una arquitectura cliente/servidor de gran rendimiento. Su desarrollo fue orientado para manejar grandes volúmenes de información, y un elevado número de transacciones. SQL Server es una aplicación completa que realiza toda la gestión relacionada con los datos. El servidor sólo tiene que enviarle una cadena de caracteres y esperar a que le devuelvan los datos.

SQL Server permite la creación de procedimientos almacenados, los cuales consisten en instrucciones SQL que se almacenan dentro de una base de datos, se trata de procedimientos que se guardan semicompilados en el servidor y que pueden ser invocados desde el cliente. Se ejecutan más rápido que instrucciones SQL independientes.

SQL Server puede manejar perfectamente bases de datos de TeraBytes con millones de registros y funciona sin problemas con miles de conexiones simultáneas a los datos.

Oracle

Oracle es un Sistema de Gestión de Bases de Datos Relacional (SGBDR). La base de datos Oracle es una herramienta muy confiable y segura, tiene opciones de auditoria, backup y aplicaciones para la toma de decisiones que la diferencian de sus competidores libres y propietarios.(PROAÑO 2006)

Es la base de datos más utilizada actualmente. El acceso a los datos se otorga según privilegios concedidos por el administrador y es posible acceder a datos de Oracle usando software de otros fabricantes. Además cuenta con procedimientos muy buenos para hacer copias de seguridad y recuperar datos.

MySQL

MySQL un sistema de gestión de base de datos relacional, multihilo y multiusuario. Es una idea originaria de la empresa open source MySQL AB. Desde enero de 2008 MySQL AB pertenece a Sun Microsystems.

MySQL AB desarrolla MySQL como software libre en un esquema de licenciamiento dual. Por un lado se ofrece bajo la GNU GPL para cualquier uso compatible con esta licencia, pero las empresas que quieran incorporarlo en productos privativos pueden comprar a la empresa una licencia específica que les permita este uso.

MySQL es muy rápido, fiable y fácil de usar, surge para manipular bases de datos muy grandes. Es un sistema multiplataforma de base de datos relacionales, lo que da velocidad y flexibilidad, cuenta con un sistema muy seguro de privilegios que permite la autenticación básica para el acceso al servidor.

El lenguaje PHP es altamente compatible con MySQL, por el amplio conjunto de comandos definidos para el tratamiento de éste.

Utiliza SQL, el lenguaje estándar para la consulta de bases de datos utilizado en todo el mundo. Existen distribuciones bajo una licencia de código abierto.

Gestor de base de datos seleccionado

Luego de analizadas las características y facilidades de los SGBD presentados, y las de la herramienta a desarrollar se decide usar el MySQL como SGBD, por las siguientes razones:

- No se necesitará de un manejo complejo de la información.
- El PHP maneja más fácil al MySQL que al SQL Server, debido a la gran cantidad de funciones que tiene explícitas.
- Bajo coste: está disponible de manera gratuita, bajo una licencia de código abierto.
- Portabilidad: se puede utilizar en gran cantidad de sistemas Unix diferentes, así como en sistemas Windows.
- Código fuente: Como en el caso de PHP, puede obtener y modificar el código fuente de MySQL.

- Según las pruebas realizadas, el motor de base de datos Mysql es mucho más rápido que Oracle en el procesamiento de transacciones. (PROAÑO 2006)

1.11 ¿Por qué usar MySQL, PHP y Apache?

Para la selección de esta combinación se tuvieron en cuenta las características que se expusieron anteriormente:

- Ofrecen un alto nivel de prestaciones en la red.
- Presentan la característica de ser multiplataforma.
- Juntos ofrecen al programador una API (Application Program Interface) muy potente que permite el desarrollo de cualquier tipo de aplicación de gestión de datos.
- Para mantener la compatibilidad con otros sistemas de gestión.
- Es relativamente fácil su uso.
- La seguridad que implementa el Apache, más la implementada por MySQL, satisfacen completamente los requerimientos de seguridad propuestos para el sistema.

1.12 UML (Unified Modeling Language)

UML (Unified Modeling Language) o Lenguaje Unificado de Modelado es un lenguaje gráfico para especificar, construir, visualizar y documentar partes o artefactos (Booch 2000).

De forma general las principales características son (Jim Octubre 04, 2002):

- Lenguaje unificado para la modelación de sistemas.
- Tecnología orientada a objetos.
- El cliente participa en todas las etapas del proyecto.
- Corrección de errores viables en todas las etapas.
- Aplicable para tratar asuntos de escala inherentes a sistemas complejos de misión crítica, tiempo real y cliente/servidor.

UML es desde finales de 1997, un lenguaje de modelado orientado a objetos estándar, de acuerdo con el Object Management Group, siendo utilizado diariamente por grandes organizaciones como: Microsoft, Oracle, Rational.

1.13 Metodologías

Conjunto de procedimientos, técnicas, herramientas que ayudan a los desarrolladores a realizar un nuevo producto.

RUP

La metodología RUP (Rational Unified Process), divide en 4 fases el desarrollo del software:

- Inicio: Determina la visión del proyecto.
- Elaboración: Determina la arquitectura óptima.
- Construcción: Obtiene la capacidad operacional inicial.
- Transición: Obtiene el release del proyecto.

Cada una de estas etapas es desarrollada mediante iteraciones, la cual consiste en reproducir el ciclo de vida en cascada a menor escala. Los objetivos de una iteración se establecen en función de la evaluación de las iteraciones precedentes.

Los elementos de RUP son:

- Actividades: Procesos que se determinan en cada iteración.
- Trabajadores: Personas o entes involucradas en cada proceso.
- Artefactos: Puede ser un documento, un modelo, o un elemento de modelo.

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.

Extreme Programming (XP)

Es una de las metodologías de desarrollo de software más exitosas en la actualidad. 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.

Características de XP, la metodología se basa en:(Sánchez 2004)

- Pruebas Unitarias: se basa en las pruebas realizadas a los principales procesos, de tal manera que adelantándose en algo hacia el futuro, se pueda hacer pruebas de las fallas que pudieran ocurrir. Es como si se adelantara la obtención de los posibles errores.
- Refabricación: se basa en la reutilización de código, para lo cual se crean patrones o modelos estándares, siendo más flexible al cambio.
- Programación en pares: una particularidad de esta metodología es que propone la programación en pares, la cual consiste en que dos desarrolladores participen en un proyecto en una misma estación de trabajo. Cada miembro lleva a cabo la acción que el otro no está haciendo en ese momento. Es como el chofer y el copiloto: mientras uno conduce, el otro consulta el mapa.

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.

MSF se compone de varios modelos encargados de planificar las diferentes partes implicadas en el desarrollo de un proyecto:

- Modelo de Arquitectura del Proyecto: Diseñado para acortar la planificación del ciclo de vida.
- Modelo de Equipo: Diseñado para mejorar el rendimiento del equipo de desarrollo.
- Modelo de Proceso: Diseñado para mejorar el control del proyecto, minimizando el riesgo, y aumentar la calidad acortando el tiempo de entrega.
- Modelo de Gestión del Riesgo: Diseñado para ayudar al equipo a identificar las prioridades, tomar las decisiones estratégicas correctas y controlar las emergencias que puedan surgir.
- Modelo de Diseño del Proceso: Diseñado para distinguir entre los objetivos empresariales y las necesidades del usuario.
- Modelo de Aplicación: Diseñado para mejorar el desarrollo, el mantenimiento y el soporte, proporciona un modelo de tres niveles para diseñar y desarrollar aplicaciones software. (Sánchez 2004)

Metodología seleccionada

La metodología propuesta para continuar con el desarrollo del sistema es RUP (Rational Unified Process) ya que es un proceso que junto con el Lenguaje Unificado de Modelado UML, constituye la metodología estándar más utilizada para el análisis, implementación y documentación de sistemas orientados a objetos. Este es un proceso que garantiza la elaboración de todas las fases de un producto de software orientado a objeto.

1.14 Rational Rose

Existen herramientas Case de trabajo visuales como el Analise, el Designe, el Rational Rose, que permiten realizar el modelado del desarrollo de los proyectos, se considera la mejor y más utilizada en el mercado mundial.

Rational Rose es la herramienta de modelación visual que provee el modelado basado en UML.

1.15 Conclusiones

A lo largo de este capítulo se ha realizado un estudio de la administración de sistemas, así como la obtención de reportes utilizando diferentes herramientas reporteadoras y a su vez se estudiaron las tendencias, técnicas, tecnologías, metodologías y software usados en la actualidad hasta llegar a la selección de las que se utilizarán para el desarrollo de los módulos administración y reportes del SAED.

Capítulo 2: Características del Sistema.

2.1 Introducción

En el presente capítulo se describen los procesos del negocio relacionados con el campo de acción, además se definen los requisitos funcionales y no funcionales que deben cumplir los módulos a desarrollar del Sistema de Análisis Estadístico de Datos (SAED). También se identifican los actores del sistema, los casos de uso y el modelo de casos de uso que describe las relaciones entre ellos.

2.2 Flujo actual de los procesos.

El Sistema de Análisis Estadístico de Datos (SAED) gestiona los procesos de elaboración y análisis estadístico de encuestas. Estos procesos brindan a los usuarios del sistema la posibilidad de elaborar las encuestas que necesiten, publicar dichas encuestas en el propio sistema o llevárselas y publicarlas en otros portales y además realizar el análisis estadístico que solicite, en caso de que el sistema no cuente con los estadígrafos solicitados por el usuario éste puede llevar los resultados de su encuesta a otro sistema. SAED permite obtener un informe con los resultados del análisis estadístico de las encuestas exportado en una tabla Excel.

El Sistema de Análisis Estadístico de Datos posee usuarios locales y las encuestas creadas por éstos permanecen en el sistema indefinidamente al igual que los usuarios aunque lleven mucho tiempo sin utilizar el sistema.

El informe que brinda el sistema no constituye un reporte personalizado o sea en él está recogido todo el análisis estadístico que se le realiza a la encuesta y no lo que verdaderamente requiere el usuario en un determinado momento. Además el hecho de que las encuestas y los usuarios inactivos permanezcan en la base de datos puede provocar lentitud en el sistema debido a la sobrecarga de ésta.

2.3 Objeto de automatización.

Para la realización de este trabajo se pretende automatizar los siguientes procesos:

- Configurar reportes seleccionando las encuestas, preguntas y estadígrafos deseados, así como la fecha en que desea ser recibido.
- Elaborar reportes a partir de la configuración realizada.
- Enviar reportes al usuario que lo solicitó.
- Consultar reportes una vez que hayan sido enviados.

2.4 Propuesta de sistema.

Para lograr el cumplimiento de los objetivos del presente trabajo se propone añadir al Sistema de Análisis Estadístico de Datos los módulos Administración y Reporte.

El módulo reporte permitirá a los usuarios configurar los reportes que necesitan, para lo cual el sistema brindará la posibilidad no solamente de escoger las encuestas, las preguntas y los estadígrafos que soliciten sino también el período de tiempo en que desea recibir el reporte.

El módulo administración permitirá gestionar las cuentas de los usuarios que la soliciten. Además una vez que el usuario haya configurado su reporte la administración se encargará de la gestión de los mismos o sea se elaborarán los reportes y se enviarán al usuario que los solicitó, el cual podrá consultarlos cuando lo desee e imprimirlos una vez que los haya exportado a formato PDF.

2.5 Modelo de negocio.

2.5.1 Descripción general de los procesos de negocio.

En el negocio existe un investigador que desea obtener un reporte de las encuestas que ha realizado, para lo que entrega al encuestador los datos que desea en el reporte, así como la fecha en la que desea recibir el mismo y éste último realiza la configuración del reporte solicitado. Esta configuración es revisada y aprobada por el investigador.

Una vez que el investigador ha revisado y aprobado la configuración, el encuestador elabora el reporte en la fecha indicada y lo entrega al investigador.

Por otro lado cuando el investigador desea consultar algún reporte de los que posee en un determinado período, informa el rango de tiempo al encuestador el cual busca los reportes y los entrega al investigador.

2.5.2 Reglas del Negocio

- El encuestador es la persona que debe configurar el reporte.
- El reporte tiene que contener el nombre de la encuesta a la que hace referencia.
- El encuestador solamente puede eliminar los reportes o encuestas personales.
- Las encuestas, cuentas de usuarios y reportes que lleven más de 6 meses inactivos en el sistema serán eliminados únicamente por el administrador.
- Solamente el administrador puede crear cuentas de usuarios.
- Solamente se pueden solicitar cuentas locales.
- Solamente se muestran los usuarios registrados con cuentas locales.

2.5.3 Representación de los casos de uso del negocio.

Tabla 2. 1 Actores del Negocio.

Actores del negocio	Justificación
Investigador	Es el encargado de iniciar la obtención y consulta de reporte y es el más beneficiado con el resultado de estos procesos de negocio.

Tabla 2. 2 Trabajadores del Negocio

Trabajadores del negocio	Justificación
Encuestador	Es la persona que se encarga de configurar el reporte y obtener el mismo. No se beneficia en ningún momento de las acciones ejecutadas en los procesos de negocio, sino que se limita a ejecutarlos.

2.5.3.1 Descripción textual de los casos de uso del negocio.

Tabla 2. 3 Descripción textual CU Negocio: Obtener Reporte

Nombre del Caso de Uso	Obtener reporte	
Actores	Investigador (inicia)	
Propósito	Permitir al Investigador la obtención de los datos que desea de determinada encuesta.	
Resumen	El caso de uso se inicia cuando el investigador le comunica al encuestador la necesidad de obtener un reporte presentándole la encuesta de la cual lo desea, así como las preguntas de las que desea el reporte con los estadígrafos requeridos, el encuestador realiza la configuración del reporte y la muestra al investigador y elabora el reporte, terminado así el caso de uso.	
Curso Normal de los eventos		
Acciones del Actor	Respuesta del proceso de negocio	
1. El investigador le comunica al encuestador la necesidad de obtener	1.1 El encuestador acepta la petición y solicita la encuesta de la que se solicita el reporte así como las preguntas a procesar para el reporte y los estadígrafos que necesita.	

un reporte.	
2. El investigador informa todos los datos al encuestador.	2.1 El encuestador revisa los datos que le entrega el investigador.
	2.2 El encuestador configura el reporte a partir de los datos entregados.
	2.3 El encuestador presenta al investigador la configuración del reporte.
3. El investigador revisa la configuración de reporte que le entrega el encuestador y la aprueba.	3.1 El encuestador oficializa la configuración.
	3.2 El encuestador elabora el reporte.
	3.4 El encuestador entrega al investigador el reporte
Curso Alternativo de los eventos	
Acción 3.a El investigador no aprueba la configuración de reporte que le propone el encuestador, solicitándole una nueva configuración. Ir a la Acción 2.	
Prioridad	Crítico
Mejoras	<ul style="list-style-type: none"> • El encuestador podrá configurar el reporte de forma automática. • El encuestador podrá elaborar el reporte de forma automática.

Tabla 2. 4 Descripción textual CU Negocio: Consultar Reporte

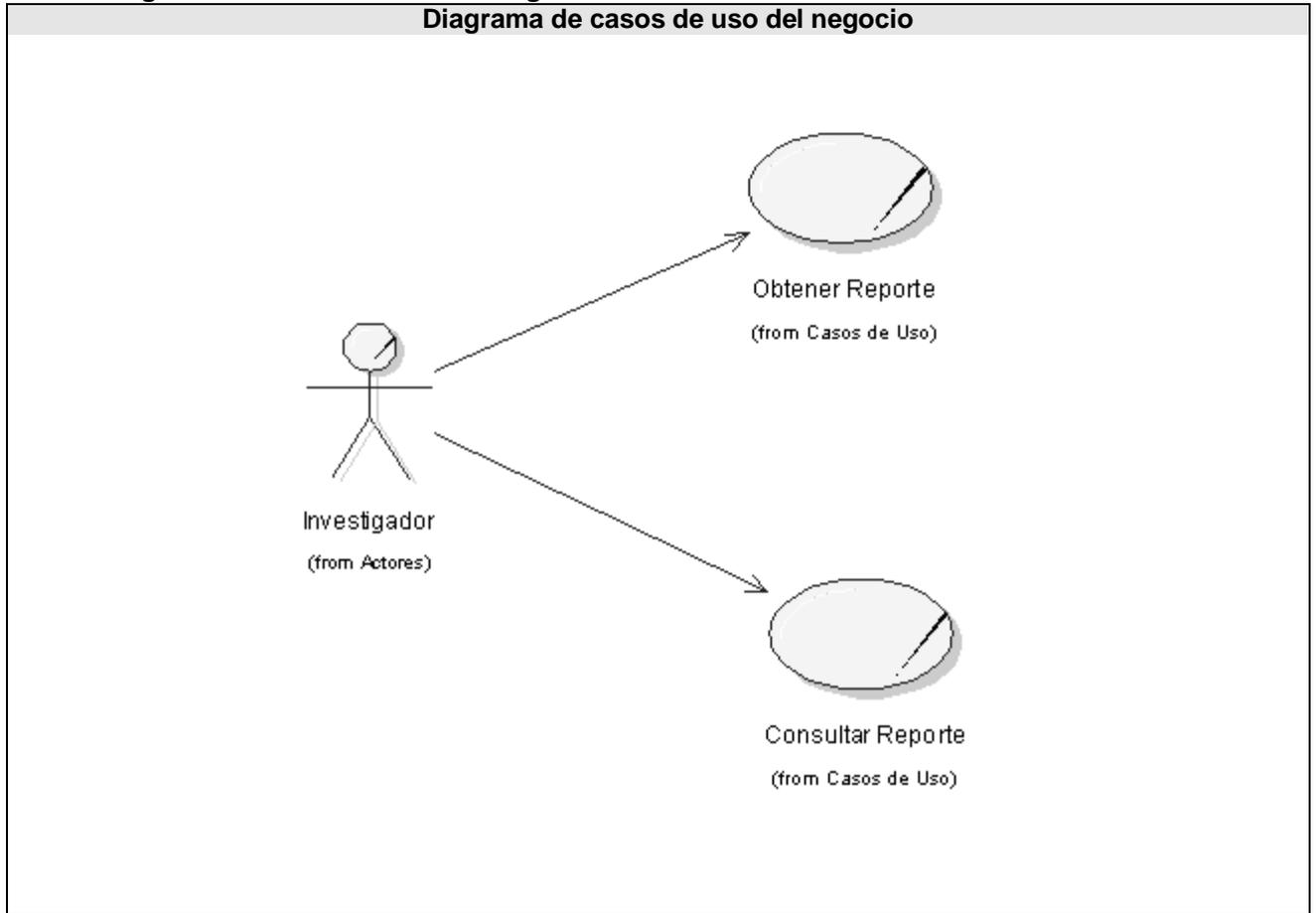
Nombre del Caso de Uso	Consultar reporte
Actores	Investigador (inicia)
Propósito	Permitir al Investigador consultar un reporte de determinada fecha

Resumen	El caso de uso se inicia cuando el investigador solicita consultar un reporte, informando la fecha del mismo, el encuestador busca el reporte de dicha fecha y se lo entrega al investigador, el cual realiza la consulta, terminado así el caso de uso.	
Curso Normal de los eventos		
Acciones del Actor	Respuesta del proceso de negocio	
1. El investigador solicita consultar un reporte.	1.1 El encuestador solicita la fecha del reporte.	
2. El investigador informa la fecha.	2.1 El encuestador busca el reporte de esa fecha.	
	2.2 El encuestador obtiene el reporte.	
	2.3 El encuestador entrega el reporte al investigador.	
3. El investigador consulta el reporte.		
Curso Alternativo de los eventos		
Acción 2.2a El encuestador no encuentra reportes en la fecha entregada, se lo informa al investigador y finaliza el caso de uso.		
Prioridad	Crítico	
Mejoras	<ul style="list-style-type: none"> • El encuestador podrá buscar el reporte de forma automática. • El encuestador podrá obtener el reporte de forma automática. 	

2.5.3.2 Diagrama de Casos de Uso del Negocio

A través del diagrama de caso de uso del negocio se representan gráficamente los procesos del negocio y su interacción con los actores del negocio.

Figura 2. 1 Diagrama de casos de Uso del Negocio.



2.5.3.3 Diagramas de actividades

Un diagrama de actividad describe un proceso que explora el orden de las tareas o actividades que logran los objetivos del negocio.

Figura 2. 2 Diagrama de actividades CU: Obtener Reporte.

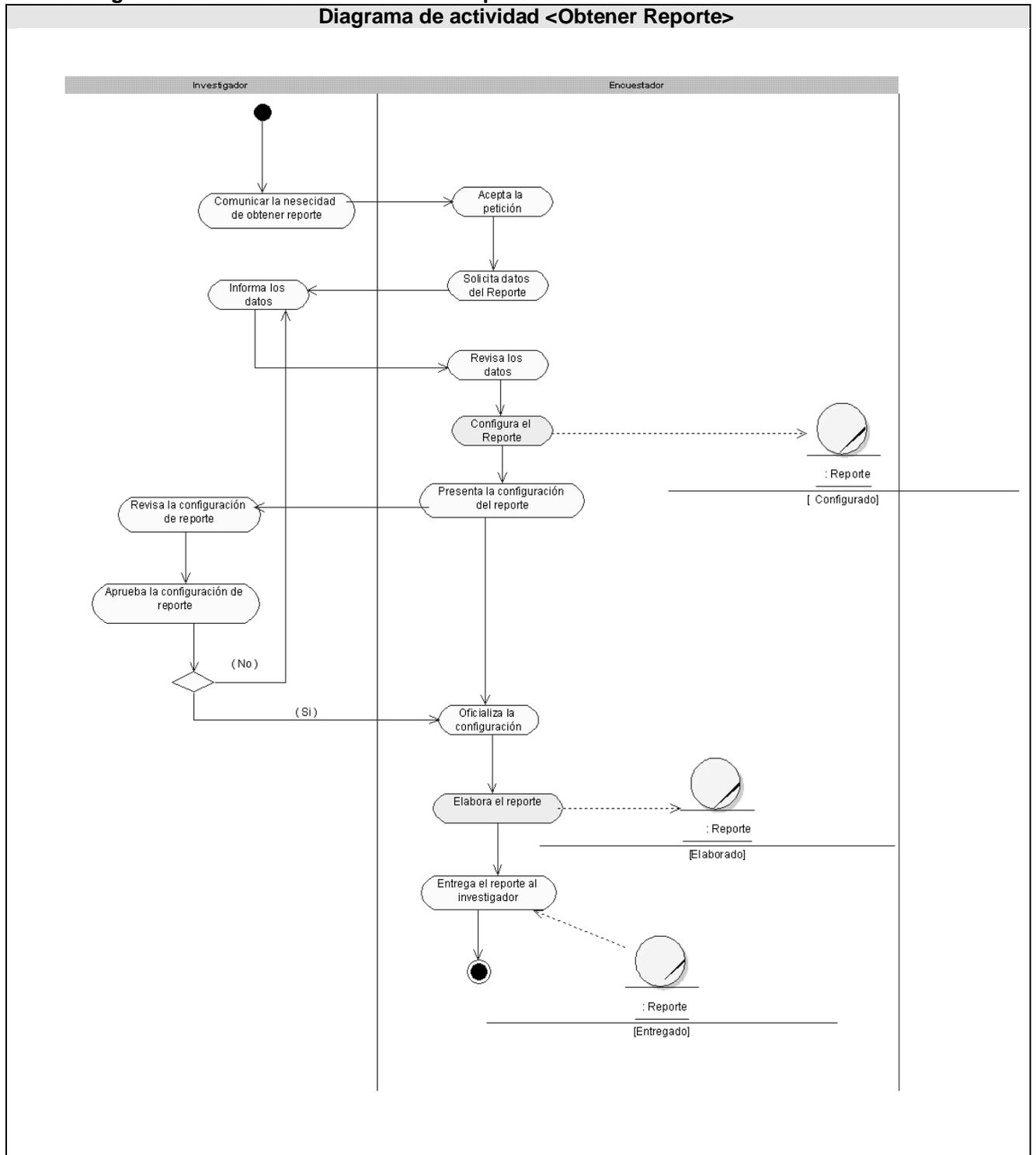
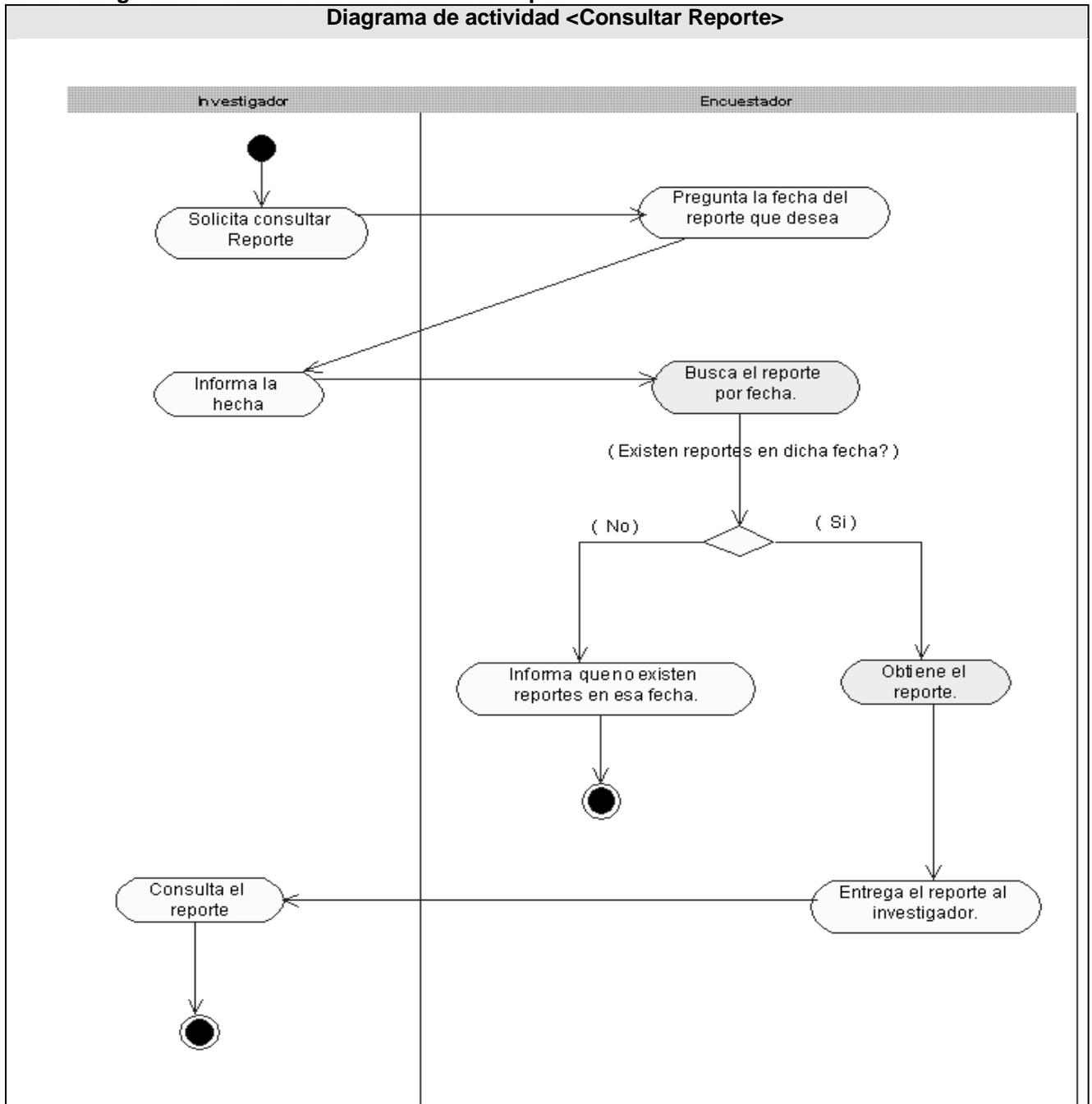


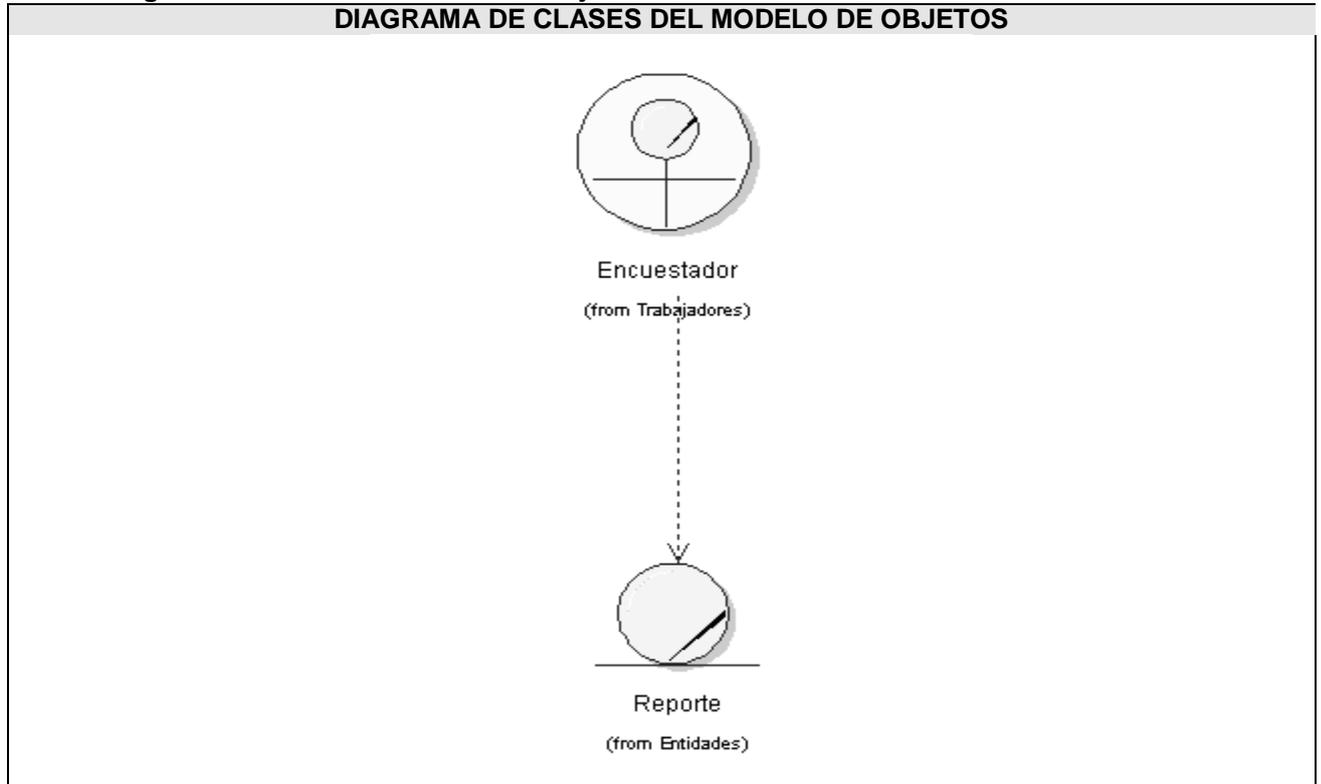
Figura 2. 3 Diagrama de actividades CU: Consultar Reporte.



2.5.4 Diagrama de clases del modelo de objetos

El diagrama de clases muestra la participación de los trabajadores y entidades del negocio y la relación entre ellos.

Figura 2. 4 Diagrama de clases del modelo de objetos.



2.6 Especificación de los requisitos de software.

2.6.1 Requerimientos Funcionales

RF1 Solicitar cuenta local.

RF 1.1 Introducir datos del usuario.

RF2 Mostrar usuarios registrados con cuentas locales.

RF3 Crear cuenta según solicitud.

RF3.1 Mostrar listado de solicitudes

RF 3.2 Eliminar solicitudes.

RF4 Modificar cuenta.

RF5 Otorgar a un usuario los privilegios del administrador

RF6 Autenticar usuario.

RF6.1 Solicitar nombre de usuario y contraseña para entrar al sistema.

RF6.2 Permitir el acceso del usuario a los recursos que le son permitidos.

RF6.3 Permitir que un usuario registrado cierre su sesión de trabajo desde cualquier parte del sistema.

RF6.4 Mostrar siempre el usuario activo.

RF7 Configurar reporte

RF7.1 Seleccionar la encuesta de la que se solicita el reporte.

RF7.2 Seleccionar las preguntas que se desea incluir en el reporte.

RF7.3 Seleccionar el o los estadígrafos que se desea aplicar.

RF7.4 Configurar la fecha en que se desea recibir el reporte.

RF8 Elaborar reporte a partir de la configuración solicitada.

RF9 Enviar reporte.

RF10 Eliminar elementos inactivos.

RF11 Consultar reporte

RF11.1 Mostrar un listado de reportes según un rango de tiempo seleccionado.

RF 11.2 Mostrar el contenido de un reporte.

RF 12 Eliminar elementos innecesarios.

2.6.2 Requerimientos no funcionales.

RNF1 Requerimiento de software

1.1 Servidor Web Apache 2.2.3.

1.2 MySql 5.027.

1.3 PHP 5.2.0.

1.4 El cliente debe tener instalado navegador compatible o superior con Internet Explorer 4, que tenga habilitado el uso de Javascript.

RNF2 Requerimiento de hardware

2.1 El Servidor debe tener como mínimo las siguientes características de hardware: Procesador Pentium II 450 MHz o superior, 128 Mb de memoria RAM (incluye la utilizada por el SO) y 2Gb de capacidad en disco duro.

2.2 Las computadoras cliente, procesador Pentium III con 64 Mb de memoria RAM como mínimo.

RNF3 Apariencia o interfaz externa

- 3.1 Diseño sencillo e intuitivo, con pocas entradas, de forma tal que facilite el uso a los usuarios que no tengan entrenamiento previo en sistemas con estas características.
- 3.2 Uso de mensajes en cada momento que contengan la información que se necesite.
- 3.3 Cada usuario tiene acceso sólo a la información que él procesa.

RNF4 Interfaz interna

- 4.1 El sistema no posee interfaces a componentes de otras aplicaciones, por lo que no posee interfaz interna con ningún componente, estándar o sistema en específico.

RNF5 Usabilidad

- 5.1 El sistema podrá ser usado por cualquier tipo de personas que posean conocimientos básicos en el manejo de la computadora y el ambiente Web en sentido general.

RNF6 Rendimiento

- 6.1 El sistema necesita tiempo de procesamiento de información y tiempo de respuestas rápidas, manteniendo siempre la consistencia de los datos.

RNF7 Portabilidad

- 7.1 Puede funcionar en cualquier sistema operativo ya que tanto el servidor web como el de base de datos son multiplataforma.

RNF8 Requerimiento de Soporte

- 8.1 Servidor de BD que soporte grandes volúmenes de datos y que tenga buena velocidad de procesamiento, con tiempo de respuesta rápida en accesos concurrentes.
- 8.2 Versión de PHP 5.2.0.

RNF9 Requerimientos de Seguridad

- 9.1 La información debe ser confidencial, para ello se establece un sistema de permisos y usuarios para el acceso a la información. Para mantener la integridad en el mismo sólo se podrá acceder al sistema después de autenticarse encriptándose la contraseña directamente en la máquina cliente utilizando el algoritmo de encriptación md5, por lo que viaja hacia el servidor de forma protegida.

RNF10 Confiabilidad

10.1 La herramienta de implementación a utilizar tiene soporte para recuperación ante fallos y errores.

10.2 El sistema debe ser capaz de mantener la integridad de los datos

2.6.3 Definición de los casos de uso.

2.6.3.1 Definición de los actores del sistema.

Tabla 2. 5 Actores del Sistema

Actores	Justificación
Encuestador	Es el encargado de configurar los reportes así como consultar los mismos.
Administrador	Se encarga de la gestión de cuentas, además de eliminar los elementos que lleven mucho tiempo inactivos en el sistema.
Usuario Anónimo	Es cualquier usuario que llegue al sistema antes de autenticarse, éste puede solicitar una cuenta para acceder al sistema.
Usuario Registrado	Este actor es cualquier usuario que llegue al sistema y posea una cuenta, éste realiza la acción de autenticarse.

2.6.3.2 Descripción textual de los casos de uso

Tabla 2. 6 Descripción textual CU Sistema: Solicitar Cuenta

SOLICITAR CUENTA	
ACTORES	Usuario anónimo (inicia)
PROPÓSITO	Permitir al usuario anónimo solicitar una cuenta para acceder al sistema
RESUMEN	El caso de uso inicia cuando un usuario requiere una cuenta para entrar al sistema. Para ello debe introducir los datos solicitados, el sistema muestra un mensaje indicando que su solicitud ha sido adicionada y así finaliza el caso de uso.
REFERENCIAS	RF 1, RF 1.1
PRECONDICIONES	El Usuario anónimo seleccionó la opción de registrarse en el sistema.

ACCIÓN DEL ACTOR		RESPUESTA DEL SISTEMA
1. El usuario anónimo selecciona la opción de Registrarse.		1.1 El sistema muestra los datos que deben ser introducidos por el usuario.
2. El usuario anónimo introduce los datos necesarios y los envía.		2.1 El sistema muestra un mensaje indicando que la solicitud ha sido adicionada.
CURSO ALTERNATIVO DE LOS EVENTOS		
ACCIÓN DEL ACTOR		RESPUESTA DEL SISTEMA
2a El usuario anónimo no llena todos los datos obligatorios y presiona el botón Enviar.		3.1 El sistema muestra un mensaje de error indicando que debe llenar todos los datos obligatorios.
2b El usuario anónimo llena todos los datos obligatorios, pero éstos son incorrectos y presiona el botón Enviar.		3.1 El sistema muestra un mensaje de error indicando que los datos son incorrectos y que debe llenarlos correctamente.
2c El usuario anónimo decide que no desea solicitar una cuenta y presiona el botón Cancelar.		3.1 El sistema, redirecciona al usuario anónimo a la página de inicio.
PRIORIDAD	Crítico	
POSCONDICIONES	El usuario ha solicitado su cuenta.	

Tabla 2. 7 Descripción textual CU Sistema: Autenticar Usuario.

AUTENTICAR USUARIO	
ACTORES	Usuario registrado
PROPÓSITO	Permitir al usuario registrado entrar al sistema.
RESUMEN	El CU inicia cuando el usuario registrado solicita entrar al sistema, por lo que necesita previa autenticación, finalizando de esta forma el CU.
REFERENCIAS	RF 6, RF 6.1, RF 6.2, RF 6.3, RF 6.4
PRECONDICIONES	El usuario registrado accedió al sistema
ACCIÓN DEL ACTOR	
1. El usuario registrado desea entrar al sistema, para lo cual debe introducir usuario y contraseña.	
RESPUESTA DEL SISTEMA	
1.1 Los datos entrados coinciden con los registrados en la base de datos, el sistema muestra la página de bienvenida al módulo de configuración, terminando el CU.	
2. El usuario registrado selecciona la opción cerrar sesión de trabajo.	
2.1 El sistema cierra la sesión y redirecciona al usuario registrado a la página inicial.	
CURSO ALTERNATIVO DE LOS EVENTOS	
ACCIÓN DEL ACTOR	
1.1a Los datos usuario y contraseña no coinciden con los registrados en la base de datos, el sistema muestra un mensaje de error especificando que no tiene acceso al sistema.	

PRIORIDAD	Crítico
POSCONDICIONES	El usuario ha entrado al sistema.

Tabla 2. 8 Descripción textual CU Sistema: Configurar Reporte.

CONFIGURAR REPORTE	
ACTORES	Encuestador (inicia)
PROPÓSITO	Configurar los datos de un reporte.
RESUMEN	El caso de uso inicia cuando un encuestador solicita la configuración de un reporte, el sistema le brinda la posibilidad de realizar dicha configuración y envía la misma al administrador, finalizando así el caso de uso
REFERENCIAS	RF 7, RF 7.1, RF 7.2, RF 7.3, RF 7.4
PRECONDICIONES	El encuestador solicitó la opción de Configurar Reporte
ACCIÓN DEL ACTOR	
RESPUESTA DEL SISTEMA	
1. El encuestador selecciona la opción de Configurar Reporte.	1.1 El sistema muestra los datos que puede seleccionar el encuestador para configurar su reporte.
2. El encuestador selecciona los datos que desea y presiona el botón Configurar.	2.1 El sistema guarda dichos datos y redirecciona al encuestador a la página de inicio, finalizando de esta forma el caso de uso.
CURSO ALTERNATIVO DE LOS EVENTOS	
ACCIÓN DEL ACTOR	
RESPUESTA DEL SISTEMA	
2a El encuestador, decide que no desea configurar el reporte y presiona el botón Cancelar.	2.1 El sistema, redirecciona al encuestador a la página de inicio.
PRIORIDAD	Crítico
POSCONDICIONES	El encuestador ha configurado su reporte

Tabla 2. 9 Descripción textual CU Sistema: Consultar Reporte.

CONSULTAR REPORTE	
ACTORES	Encuestador (inicia)
PROPÓSITO	Consultar los reportes que desee el encuestador.
RESUMEN	El caso de uso inicia cuando un encuestador solicita consultar sus reportes, el sistema le brinda la posibilidad de seleccionar los reportes que desea, se los muestra y finaliza el caso de uso.
REFERENCIAS	RF 11, RF 11.1, RF 11.2
PRECONDICIONES	El encuestador seleccionó la opción de Consultar Reporte
ACCIÓN DEL ACTOR	
RESPUESTA DEL SISTEMA	
1. El encuestador selecciona la opción de Consultar Reporte	1.1 El sistema le ofrece la posibilidad de escoger los reportes en un rango de tiempo.

2. El encuestador selecciona el rango de tiempo deseado.	2.1 El sistema muestra una lista con los reportes que posee el encuestador en el rango de tiempo seleccionado.
3. El encuestador selecciona cada uno de los reportes que desea consultar en la lista.	3.1 El sistema muestra los datos de los reportes que selecciona el encuestador finalizando así el caso de uso
CURSO ALTERNATIVO DE LOS EVENTOS	
ACCIÓN DEL ACTOR	RESPUESTA DEL SISTEMA
2a El encuestador selecciona un rango de tiempo en el que no tiene reportes	2.1 El sistema muestra un mensaje indicando que no hay reportes en ese período.
PRIORIDAD	Crítico
POSCONDICIONES	El encuestador ha consultado los reportes que deseaba.

Tabla 2. 10 Descripción textual CU Sistema: Eliminar Elementos Innecesarios.

ELIMINAR ELEMENTOS INNECESARIOS	
ACTORES	Encuestador
PROPÓSITO	Permitir al encuestador eliminar los reportes y encuestas que posee y que ya no le son necesarios.
RESUMEN	El CU inicia cuando un encuestador desea eliminar algún elemento que ya no le es útil para su trabajo, entonces solicita la opción de eliminarlo, el sistema lo elimina de la base de datos finalizando así el caso de uso.
REFERENCIAS	RF10
PRECONDICIONES	El encuestador está autenticado como tal.
ACCIÓN DEL ACTOR	RESPUESTA DEL SISTEMA
1. El encuestador se autentica.	1.1 El sistema muestra entre otras las opciones de Gestionar Encuesta y Gestionar Reporte.
SECCIÓN ELIMINAR REPORTE	
2. El encuestador accede a la opción Gestionar Reporte.	2.1 El sistema muestra los reportes que posee el encuestador.
3. El encuestador selecciona el reporte que desea borrar y presiona el botón eliminar.	3.1 El sistema elimina dicho reporte de la base de datos.
CURSO ALTERNATIVO DE LOS EVENTOS	
ACCIÓN DEL ACTOR	RESPUESTA DEL SISTEMA
	2.1a El encuestador no posee reportes, el sistema muestra un mensaje informándolo.
SECCIÓN ELIMINAR ENCUESTAS	
2. El encuestador accede a la opción Gestionar Encuesta.	2.1 El sistema muestra las encuestas que posee el encuestador.
3. El encuestador selecciona la encuesta que desea borrar y presiona el botón	3.1 El sistema elimina dicha encuesta de la base de datos.

eliminar.	
CURSO ALTERNATIVO DE LOS EVENTOS	
	2.1a El encuestador no posee ninguna encuesta, entonces el sistema muestra un mensaje informándoselo.
PRIORIDAD	Crítico
POSCONDICIONES	Se ha eliminado un elemento innecesario del sistema.

Tabla 2. 11 Descripción textual CU Sistema: Enviar Reporte

ENVIAR REPORTE	
ACTORES	Encuestador.
PROPÓSITO	Entregar a cada usuario los reportes solicitados por él
RESUMEN	El caso de uso inicia cuando el encuestador se autentica, el sistema le muestra un cartel indicándole que tiene reportes por revisar, el sistema elabora el reporte y lo envía al encuestador, finalizando así el caso de uso.
REFERENCIAS	RF8, RF9
PRECONDICIONES	El encuestador se ha autenticado.
ACCIÓN DEL ACTOR	
RESPUESTA DEL SISTEMA	
1. El encuestador se autentica	1.1 El sistema revisa si existe algún reporte que tenga una fecha igual o anterior a la del momento en que se autenticó.
	1.2 El sistema muestra un cartel indicando que hay reportes por revisar.
2. El encuestador solicita ver el reporte.	2.1 El sistema elabora el reporte y lo muestra en formato PDF.
CURSO ALTERNATIVO DE LOS EVENTOS	
2 a El encuestador solicita ver su reporte en otro momento.	2.1 a El sistema genera el reporte en formato PDF y lo guarda en el buzón del usuario.
PRIORIDAD	Crítico
POSCONDICIONES	Se generó un Reporte en formato PDF.

Tabla 2. 12 Descripción textual CU Sistema: Eliminar Elementos Inactivos.

ELIMINAR ELEMENTOS INACTIVOS	
ACTORES	Administrador
PROPÓSITO	Permitir al administrador eliminar tanto las cuentas como los reportes y encuestas que lleven más de 6 meses inactivos en el sistema.
RESUMEN	El CU inicia cuando el administrador debe eliminar algún elemento en la base de datos que lleve más de 6 meses inactivo, para ello escoge

	los elementos y los elimina, finalizando así el caso de uso.
REFERENCIAS	RF 10
PRECONDICIONES	El administrador seleccionó la opción Administrar.
ACCIÓN DEL ACTOR	RESPUESTA DEL SISTEMA
1. El administrador accede a la página de administración.	1.1 El sistema muestra las opciones que éste posee.
SECCIÓN ELIMINAR CUENTA	
2. El administrador accede a la sección Eliminar Cuenta.	2.1 El sistema muestra las cuentas de los usuarios que no se hayan autenticado por más de 6 meses.
3 El administrador selecciona dichas cuentas y presiona el botón eliminar.	3.1 El sistema elimina dichas cuentas de la base de datos.
CURSO ALTERNATIVO DE LOS EVENTOS	
ACCIÓN DEL ACTOR	RESPUESTA DEL SISTEMA
	2.1a No existen cuentas de usuarios con más de 6 meses de inactividad, el sistema muestra un mensaje informándolo al administrador.
SECCIÓN ELIMINAR ENCUESTAS	
2. El administrador accede a la sección Eliminar Encuesta	2.1 El sistema muestra las encuestas que lleven más de 6 meses sin ser publicadas
3. El administrador selecciona dichas encuestas y presiona el botón eliminar.	3.1 El sistema elimina dichas encuestas de la base de datos.
CURSO ALTERNATIVO DE LOS EVENTOS	
	2.1a No existen encuestas que se hayan dejado de publicar por más de 6 meses, entonces el sistema muestra un mensaje informándolo al administrador.
SECCIÓN ELIMINAR REPORTE	
2. El administrador accede a la sección Eliminar Reporte	2.1 El sistema muestra los reportes que lleven más de 6 meses sin ser consultados.
3. El administrador selecciona dichos reportes y presiona el botón eliminar.	3.1 El sistema elimina dichos reportes de la base de datos
CURSO ALTERNATIVO DE LOS EVENTOS	
	2.1a No existen reportes con más de 6 meses sin ser consultados, entonces el sistema muestra un mensaje informándolo al administrador.
PRIORIDAD	Crítico
POSCONDICIONES	Se han eliminado las cuentas, encuestas y reportes que llevan más de 6 meses inactivos en el sistema.

Tabla 2. 13 Descripción textual CU Sistema: Crear Cuenta.

CREAR CUENTA	
ACTORES	Administrador (inicia)
PROPÓSITO	Crear una cuenta local al usuario que la solicitó.
RESUMEN	El caso de uso inicia cuando el administrador debe crear una cuenta a partir de la solicitud de un usuario, el sistema muestra un listado de las solicitudes y éste las acepta finalizando así el caso de uso.
REFERENCIAS	RF 3, RF 3.1, RF 3.2, RF 5
PRECONDICIONES	El administrador seleccionó la opción Administrar.
ACCIÓN DEL ACTOR	
RESPUESTA DEL SISTEMA	
1. El administrador accede a la página de administración.	1.1 El sistema muestra las opciones que éste posee.
2. El administrador selecciona la opción Registrar usuario.	2.1 El sistema muestra una lista con las solicitudes que existen.
3. El administrador acepta la solicitud.	3.1 El sistema adiciona el usuario a la base de datos.
	3.2 El sistema elimina las solicitudes de los usuarios a los que se les creó la cuenta
CURSO ALTERNATIVO DE LOS EVENTOS	
ACCIÓN DEL ACTOR	
RESPUESTA DEL SISTEMA	
	2.1 a No existen solicitudes, entonces el sistema muestra un cartel informándolo al administrador.
PRIORIDAD	Crítico
POSCONDICIONES	Se ha creado una cuenta local en el sistema.

Tabla 2. 14 Descripción textual CU Sistema: Editar Perfil.

EDITAR PERFIL	
ACTORES	Usuario Registrado (inicia)
PROPÓSITO	Modificar los datos de algún usuario.
RESUMEN	El caso de uso inicia cuando cualquier usuario registrado selecciona la opción de modificar su perfil, el sistema muestra los datos que puede modificar, éste realiza los cambios y el sistema los guarda, finalizando así el caso de uso
REFERENCIAS	RF 4
PRECONDICIONES	El usuario registrado solicitó la opción Editar Perfil.
ACCIÓN DEL ACTOR	
RESPUESTA DEL SISTEMA	
1. El usuario registrado selecciona la opción de Editar Perfil	1.1 El sistema muestra los datos que pueden ser modificados
2. El usuario registrado realiza los cambios deseados y presiona el botón	2.1 El sistema guarda dichos datos y redirecciona al usuario registrado a la página de inicio, finalizando de

Aceptar.	esta forma el caso de uso.
CURSO ALTERNATIVO DE LOS EVENTOS	
ACCIÓN DEL ACTOR	RESPUESTA DEL SISTEMA
2a El usuario registrado, decide que no desea editar su perfil y presiona el botón Cancelar.	2.1 a El sistema, redirecciona al usuario registrado a la página de inicio.
	2.1 b Los datos introducidos por el usuario son incorrectos, el sistema muestra un mensaje informándolo y redirecciona al usuario registrado a la página de inicio
PRIORIDAD	Secundario
POSCONDICIONES	El usuario registrado ha modificado su perfil.

Tabla 2. 15 Descripción textual CU Sistema: Mostrar Usuarios Registrados.

MOSTRAR USUARIOS REGISTRADOS	
ACTORES	Usuario Anónimo (inicia)
PROPÓSITO	Mostrar una lista de los usuarios registrados en el sistema con cuentas locales.
RESUMEN	El caso de uso inicia cuando cualquier usuario anónimo selecciona la opción Mostrar Usuarios Registrados para verificar si ya posee una cuenta local, el sistema muestra una lista con todas las cuentas locales que posee el sistema, finalizando así el caso de uso.
REFERENCIAS	RF 2
PRECONDICIONES	El usuario anónimo solicitó la opción Mostrar Usuarios Registrados.
ACCIÓN DEL ACTOR	
RESPUESTA DEL SISTEMA	
1. El usuario anónimo selecciona la opción de Mostrar Usuarios Registrados	1.1 El sistema muestra los cuentas locales que están registradas en el sistema.
2. El usuario revisa la lista y presiona el botón aceptar	2.1 El sistema redirecciona al usuario anónimo a la página principal, finalizando de esta forma el caso de uso.
PRIORIDAD	Secundario
POSCONDICIONES	El usuario anónimo ha consultado la lista de cuenta locales del sistema.

Tabla 2. 16 Descripción textual CU Sistema: Asignar Administración.

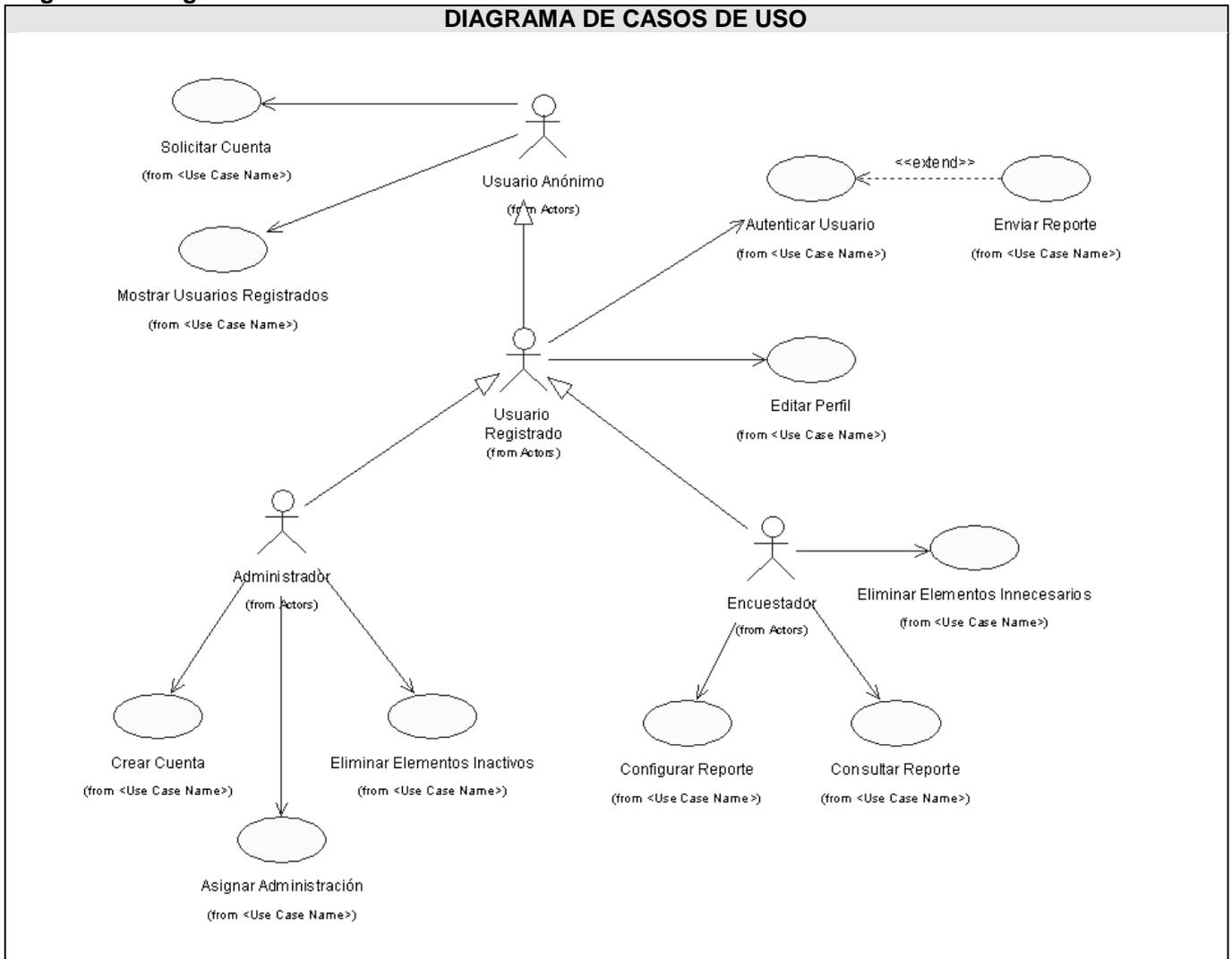
ASIGNAR ADMINISTRACIÓN	
ACTORES	Administrador (inicia)
PROPÓSITO	Otorgar a un usuario los privilegios del administrador.
RESUMEN	El caso de uso inicia cuando el administrador debe otorgar los privilegios de administración a otro usuario, para ésto el sistema muestra una lista con todos los usuarios del sistema, el administrador selecciona el usuario al que le otorgará los privilegios y el sistema guardará la modificación realizada a la cuenta de dicho usuario, finalizando así el caso de uso.
REFERENCIAS	RF 5

PRECONDICIONES	El administrador seleccionó la opción Nuevo Administrador.
ACCIÓN DEL ACTOR	
1. El administrador accede a la página de administración.	RESPUESTA DEL SISTEMA 1.1 El sistema muestra las opciones que éste posee.
2. El administrador selecciona la opción Nuevo Administrador.	2.1 El sistema muestra una lista con los usuarios que existen en el sistema.
3. El administrador selecciona el usuario al que le otorgará los privilegios.	3.1 El sistema modifica los privilegios de la cuenta de dicho usuario.
CURSO ALTERNATIVO DE LOS EVENTOS	
ACCIÓN DEL ACTOR	RESPUESTA DEL SISTEMA
3 a El administrador no desea entregar los privilegios a ningún usuario y presiona el botón cancelar.	3.1 a El sistema redirecciona al administrador a la página de administración.
PRIORIDAD	Secundario
POSCONDICIONES	Se le han otorgado los privilegios de administración a un usuario.

2.6.3.3 Diagrama de casos de uso.

Un diagrama de casos de uso del sistema representa gráficamente a los procesos y su interacción con los actores.

Figura 2. 5 Diagrama de Casos de uso del Sistema



2.7 Conclusiones

Después de haber realizado la descripción de los procesos de negocio relacionados con el campo de acción y de definir los requerimientos funcionales se obtuvo el diagrama de caso de uso del sistema en el que se mostraron los 4 actores con los que contará el SAED y además los 11 casos de uso que darán cumplimiento a todos los requisitos funcionales definidos anteriormente. También se obtuvo en este capítulo un listado de los requerimientos de software y hardware que debe cumplir el sistema.

Capítulo 3: Análisis y Diseño del Sistema.

3.1 Introducción

En este capítulo se expone el análisis y diseño del sistema. Primeramente se realiza el análisis, que representa una aproximación al diseño, con los diagramas de clases correspondientes, así como los diagramas de colaboración entre las mismas. A continuación se muestra el diseño del sistema, para ello se realizan los diagramas de clases utilizando los estereotipos Web que representan los componentes de la aplicación. Además en este capítulo se presenta el diagrama de clases persistentes y el diagrama entidad relación que constituyen la base para la construcción de la base de datos.

3.2 Análisis.

Mediante el análisis se analizan los requisitos con el objetivo de conseguir una comprensión más precisa de los mismos y una descripción que sea fácil de mantener y que ayude a estructurar todo el sistema. En este punto se puede emplear el lenguaje de los desarrolladores para describir los resultados.

3.2.1 Modelo de clases de análisis.

Los diagramas de clases del análisis muestran las clases participantes en el caso de uso y las relaciones que existen entre éstas.

Figura 3. 1 Diagrama de clases del análisis CU: Auténticar Usuario

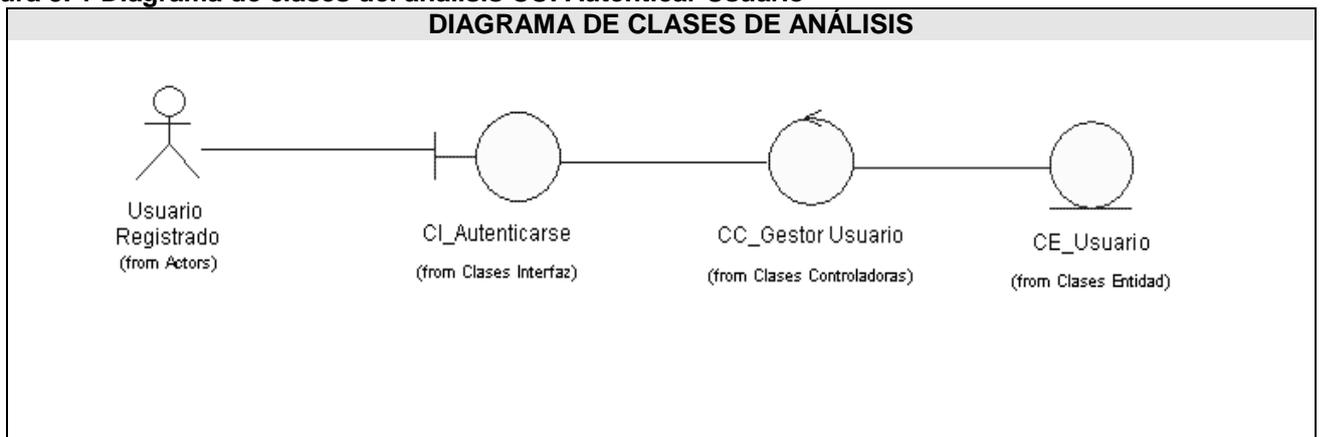


Figura 3. 2 Diagrama de clases del análisis CU: Configurar Reporte.

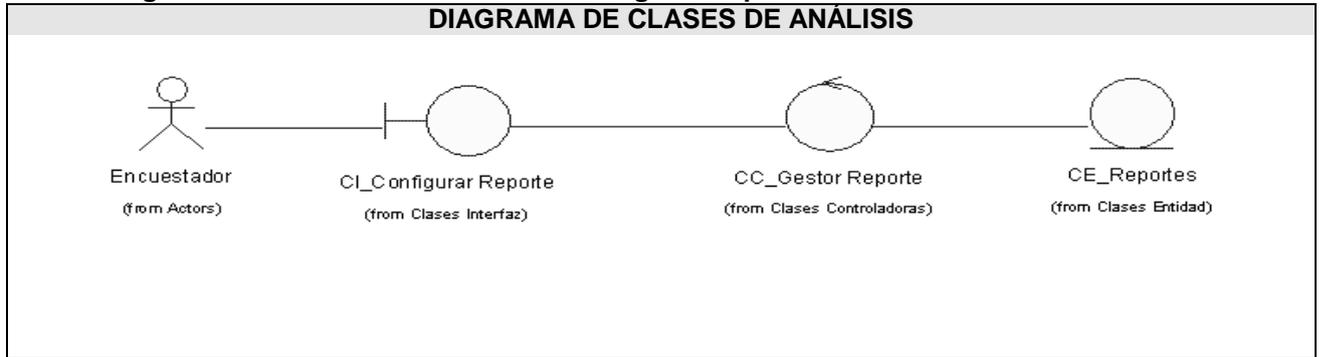


Figura 3. 3 Diagrama de clases del análisis CU: Consultar Reporte.

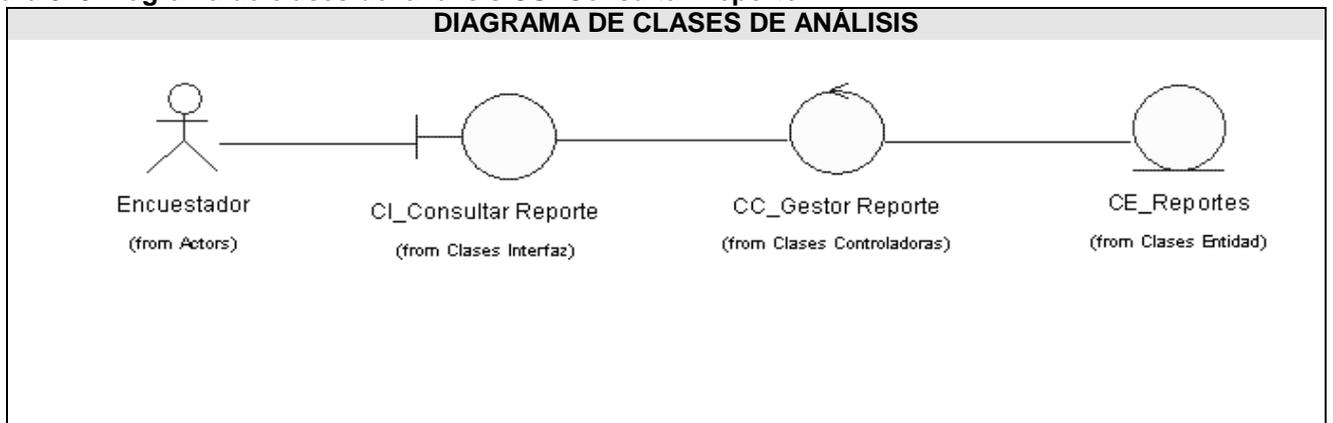


Figura 3. 4 Diagrama de clases del análisis CU: Enviar Reporte.

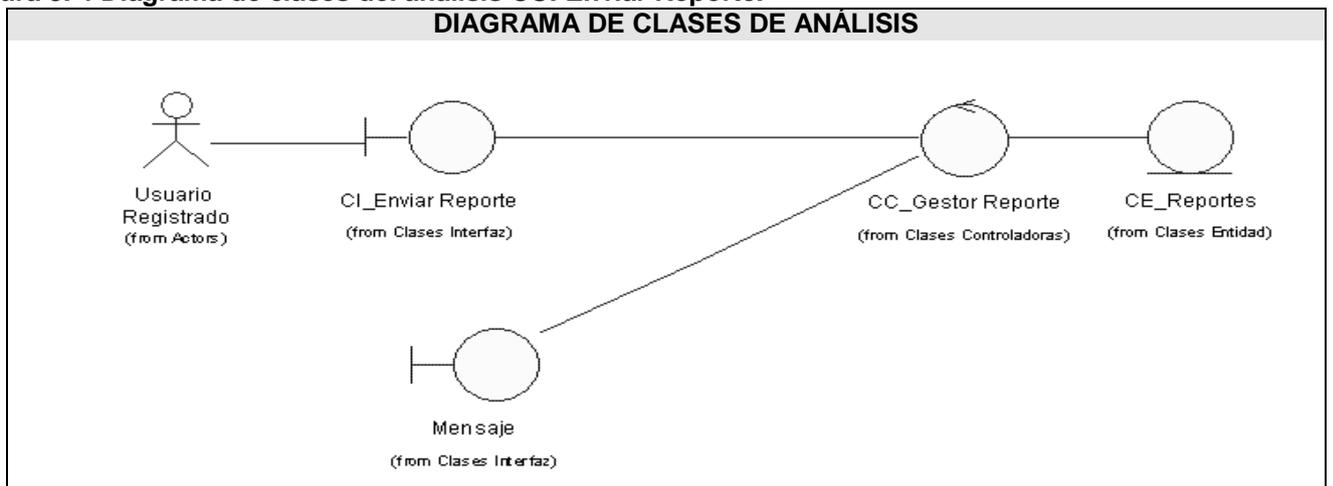


Figura 3. 5 Diagrama de clases del análisis CU: Solicitar Cuenta.

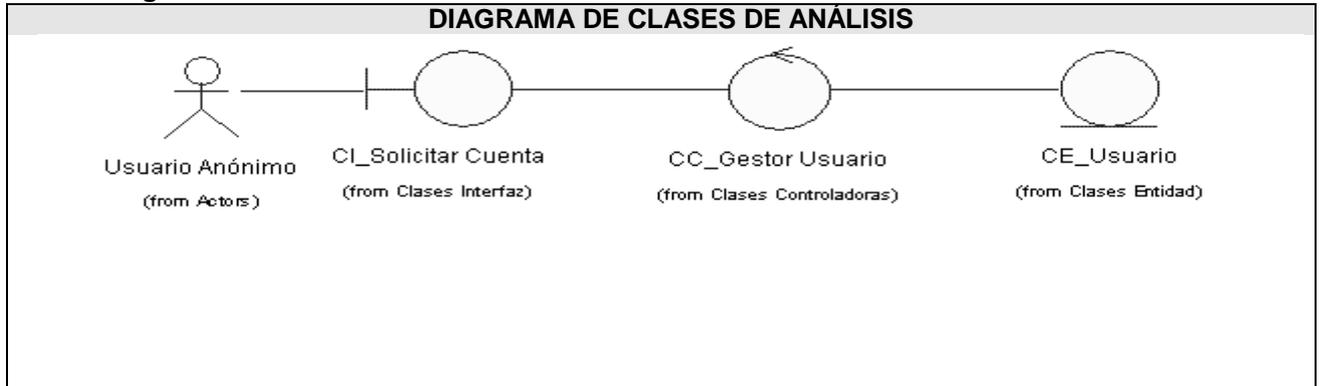


Figura 3. 6 Diagrama de clases del análisis CU: Crear Cuenta.

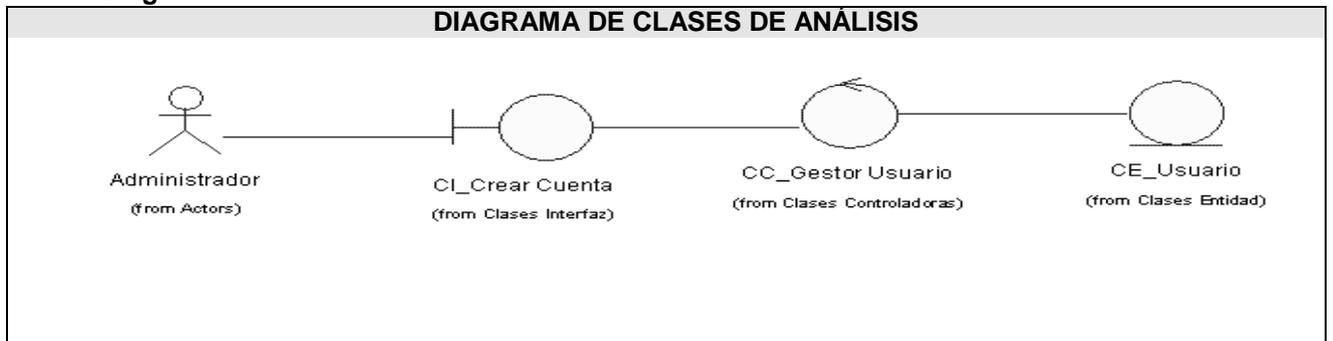


Figura 3. 7 Diagrama de clases del análisis CU: Eliminar Elementos Inactivos.

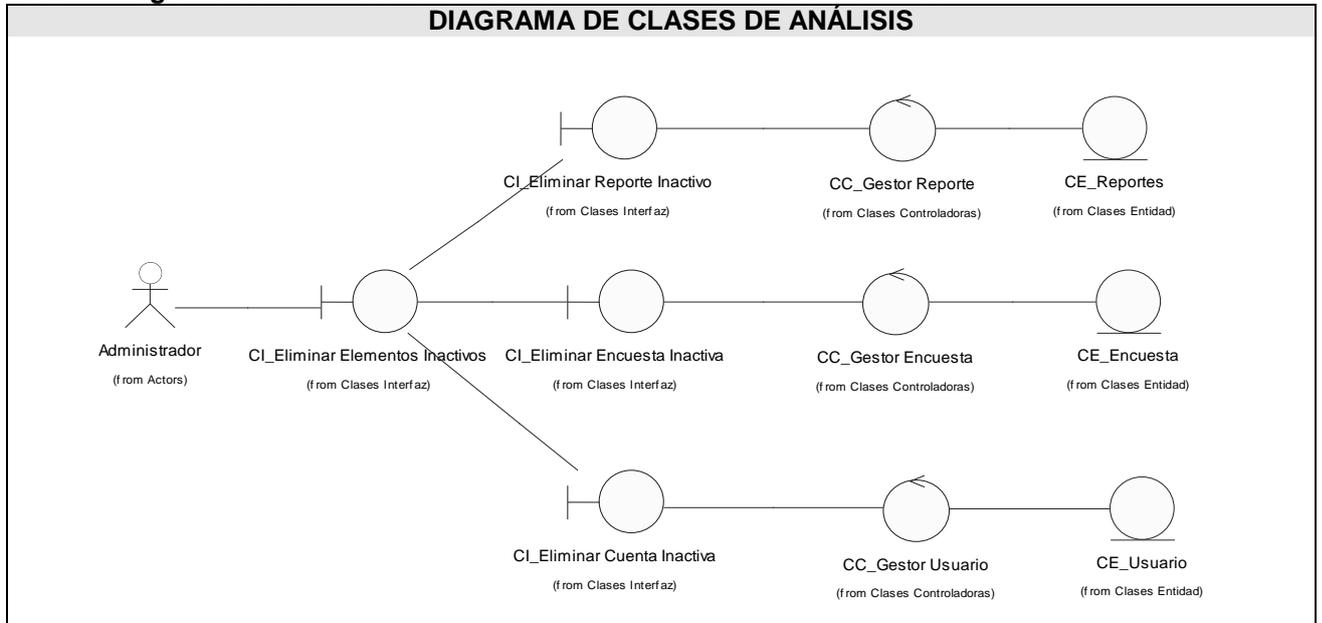


Figura 3. 8 Diagrama de clases del análisis CU: Editar Perfil.

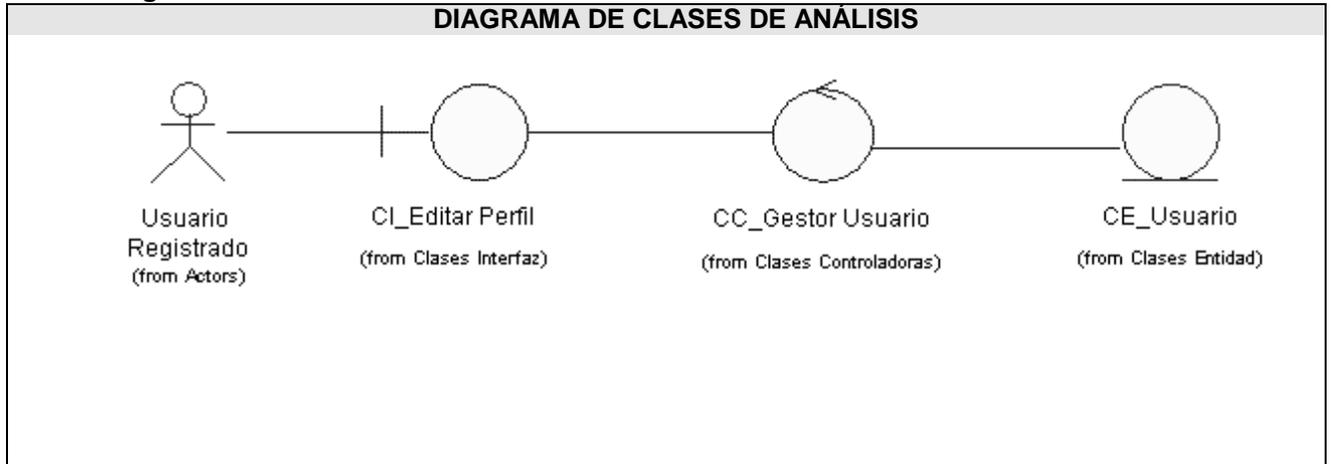


Figura 3. 9 Diagrama de clases del análisis CU: Eliminar Elementos Innecesarios.

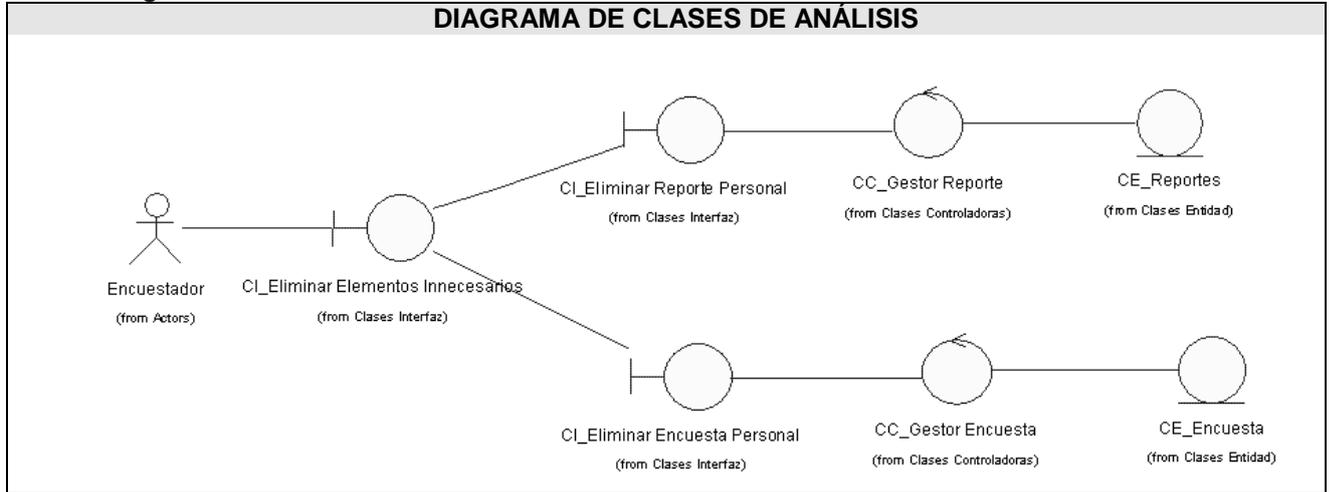


Figura 3. 10 Diagrama de clases del análisis CU: Mostrar Usuarios Registrados.

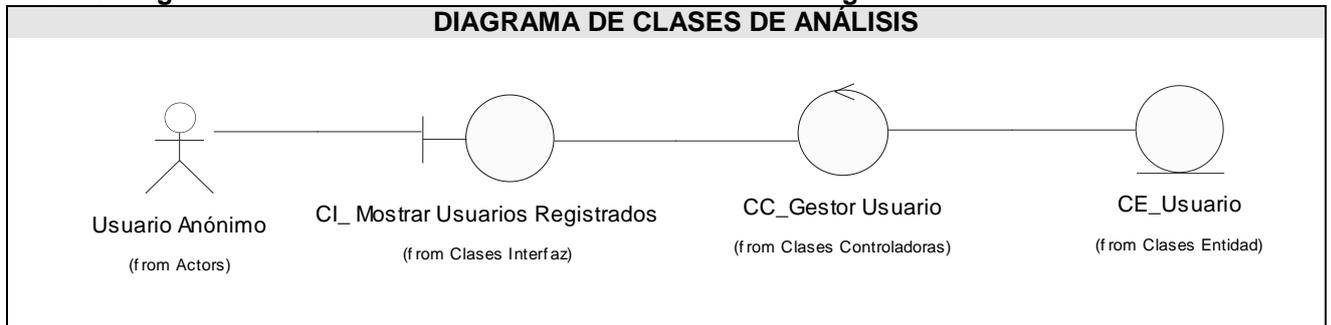
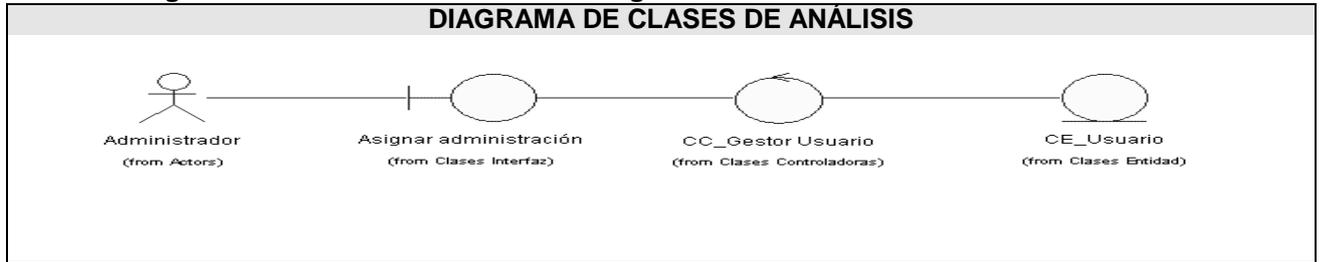


Figura 3. 11 Diagrama de clases del análisis CU: Asignar Administración.



3.2.2 Diagramas de interacción.

En el análisis es recomendable mostrar diagramas de colaboración ya que el objetivo fundamental es identificar requisitos y responsabilidades sobre los objetos y no identificar secuencias de interacción detalladas y ordenadas cronológicamente.

Se han elaborado los diagramas de colaboración correspondientes a cada uno de los escenarios de los casos de uso en estudio. (Ver anexo 1)

3.3 Diseño.

En el diseño se modela el sistema y se encuentra la forma para que soporte todos los requisitos, tanto funcionales como no funcionales. Los principales objetivos del diseño son la adquisición de una buena comprensión de los aspectos relacionados con los requisitos no funcionales y la creación de una entrada apropiada para las actividades de implementación.

3.3.1 Diseño Arquitectónico

Para la implementación del sistema propuesto, se utilizó un diseño arquitectónico en 3 capas. A continuación se muestra una definición de cada capa(LASCANO 2007):

Presentación: la capa de presentación representa la parte del sistema con la que interactúa el usuario. En una aplicación Web, un navegador puede utilizarse como cliente del sistema, pero ésta no es la única posibilidad, también puede generarse una aplicación que cumpla las funciones de un cliente “ligero” para interactuar con el usuario.

Lógica de negocio: el comportamiento de la aplicación es definido por los componentes que modelan la lógica de negocio. Estos componentes reciben las acciones a realizar a través de la capa de presentación, y llevan a cabo las tareas necesarias utilizando la capa de datos para manipular la información del sistema. Tener la lógica de negocio separada del resto del sistema también permite

una integración más sencilla y eficaz con sistemas externos, ya que la misma lógica utilizada por la capa de presentación puede ser accedida desde procesos automáticos que intercambian información con los mismos.

Acceso a datos: sus funciones incluyen el almacenamiento, la actualización y la consulta de todos los datos contenidos en el sistema. En la práctica, esta capa es esencialmente un servidor de bases de datos aunque podría ser cualquier otra fuente de información. Gracias a esta división, es posible agregar soporte para una nueva base de datos en un período de tiempo relativamente corto. La capa de datos puede estar en el mismo servidor que las de lógica de negocio y presentación, en un servidor independiente, o incluso estar distribuida entre un conjunto de servidores.

3.3.2 Diagramas de interacción.

En el diseño, es preferible la realización de diagramas de secuencias debido a que lo que se persigue es encontrar secuencias de interacciones detalladas y ordenadas en el tiempo.

Se han elaborado los diagramas de secuencias correspondientes a cada uno de los escenarios de los casos de uso en estudio. (Ver anexo 2)

3.3.3 Diagramas de clases

Los diagramas de clases del diseño muestran las clases participantes en el caso de uso, subsistemas y sus relaciones.

Figura 3. 12 Diagrama de clases del diseño CU: Autenticar Usuario.

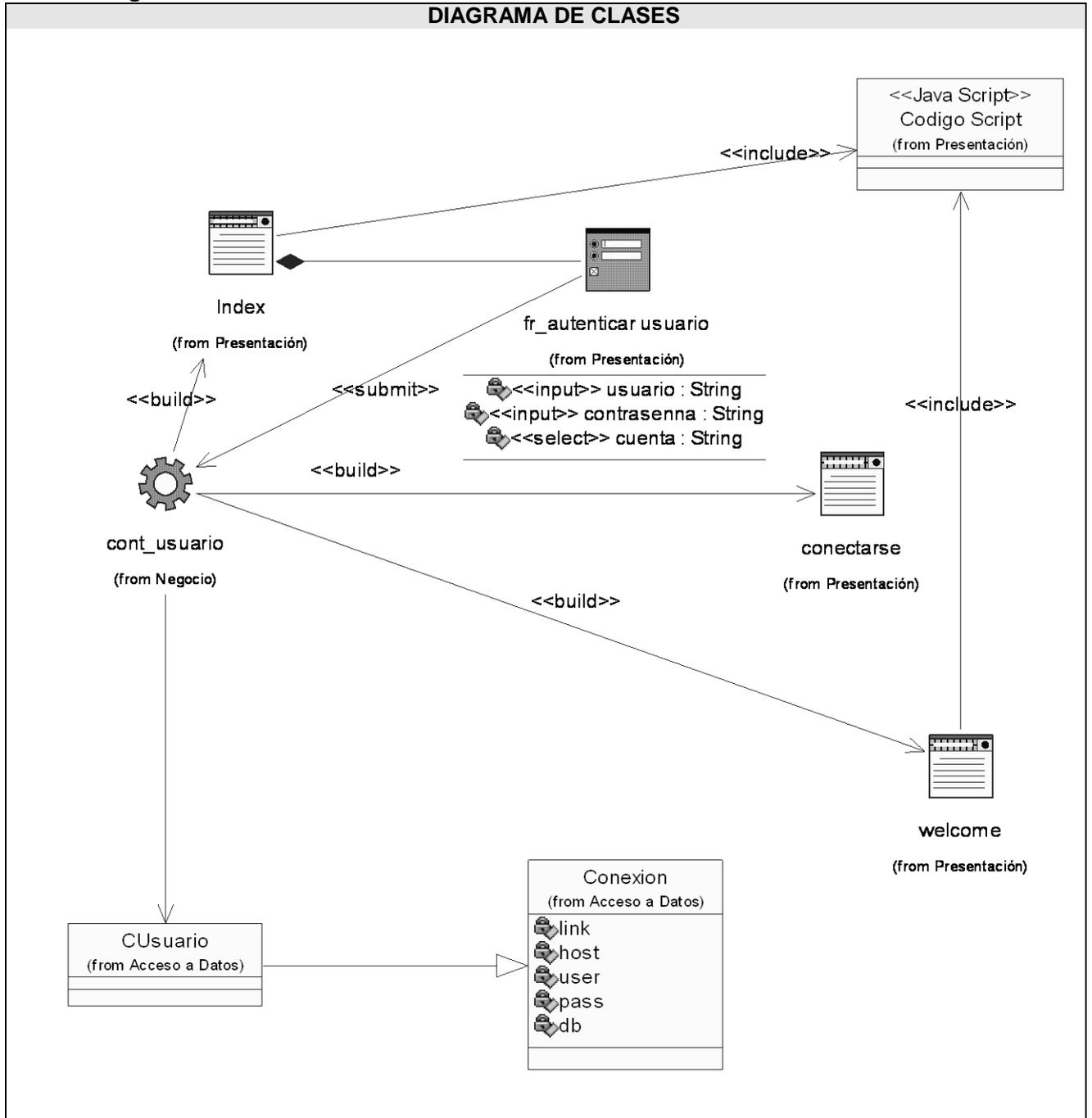


Figura 3. 13 Diagrama de clases del diseño CU: Crear Cuenta.

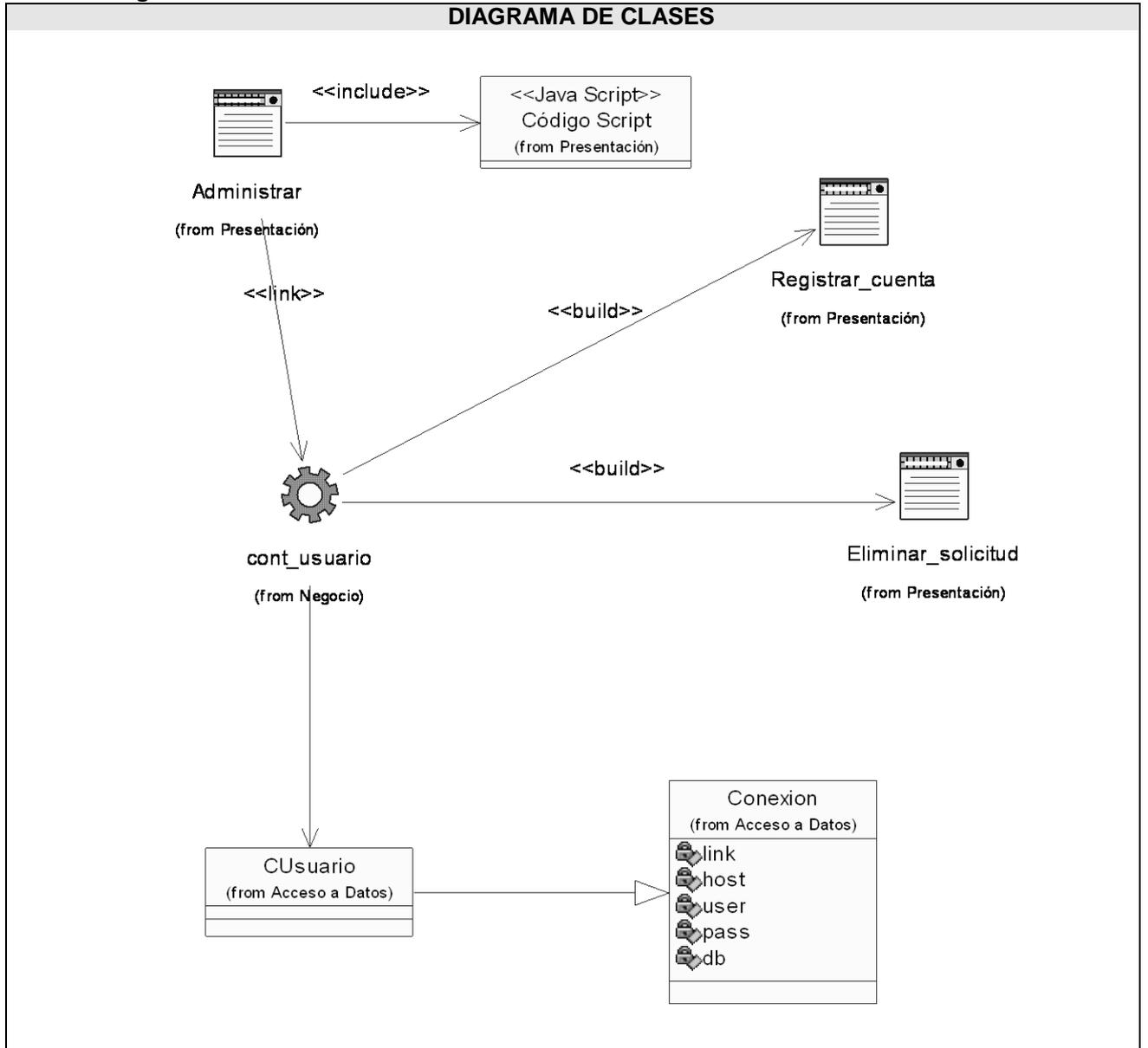


Figura 3. 14 Diagrama de clases del diseño CU: Editar Perfil.

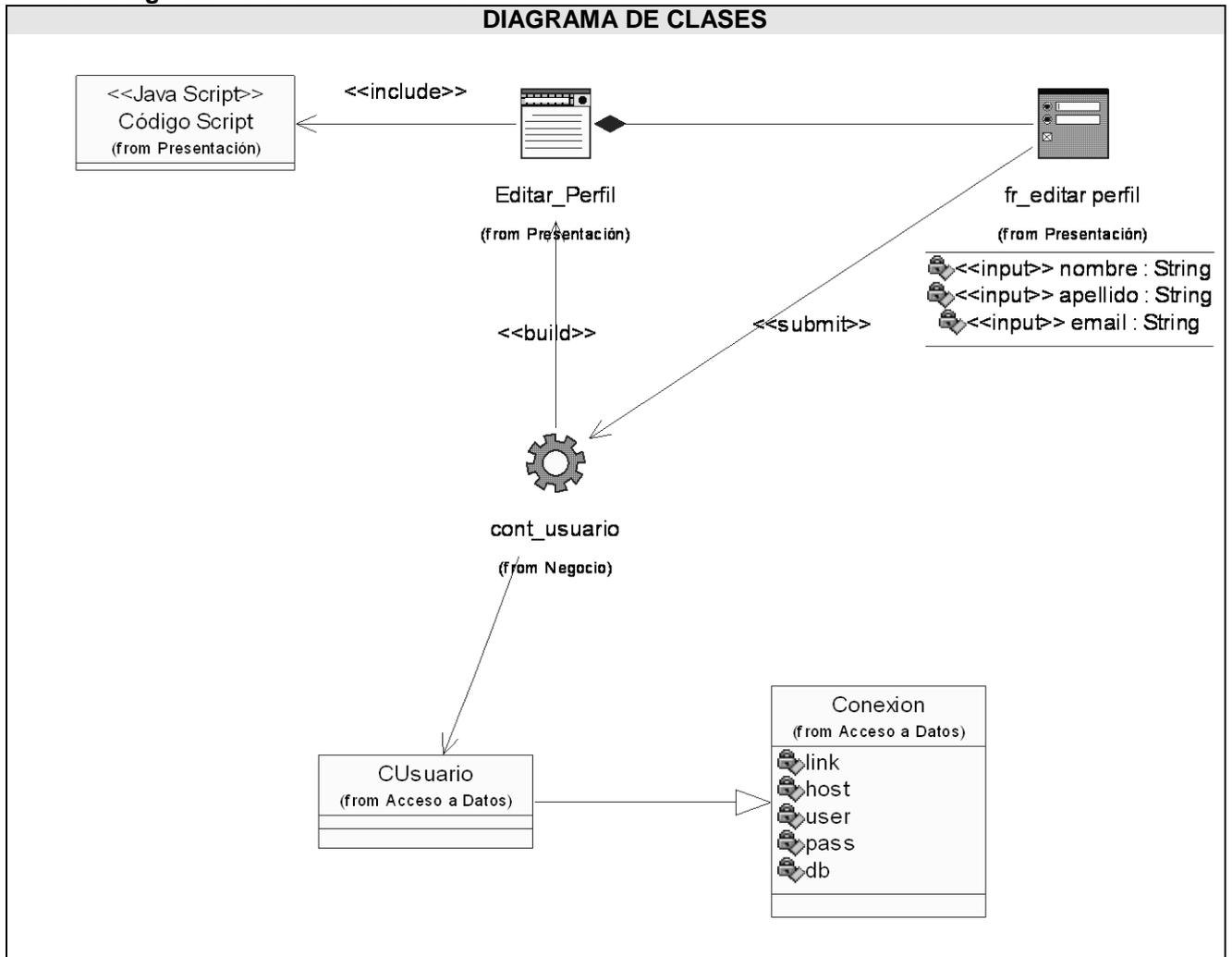


Figura 3. 15 Diagrama de clases del diseño CU: Asignar Administración.

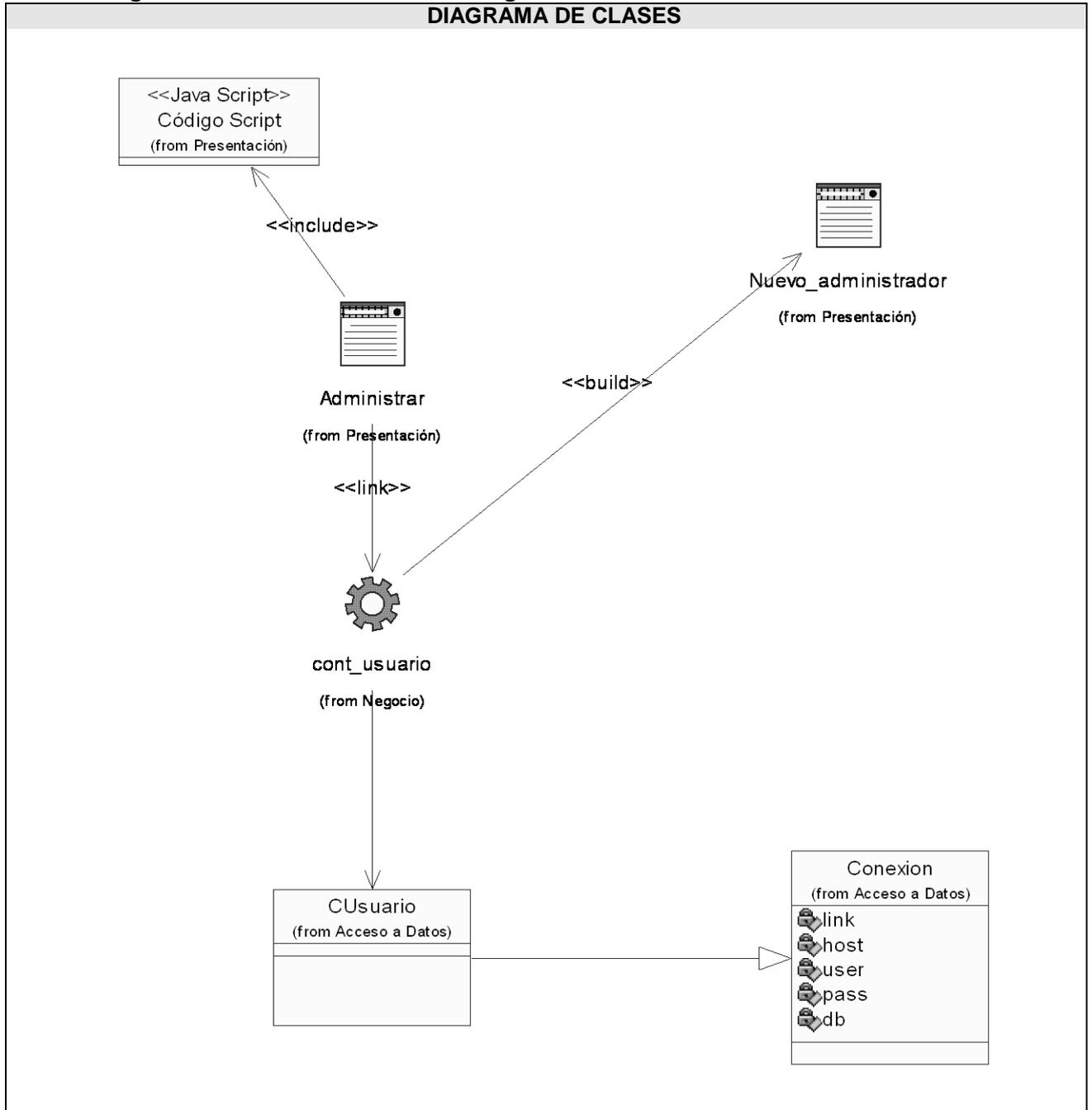


Figura 3. 16 Diagrama de clases del diseño CU: Eliminar Elementos Inactivos.

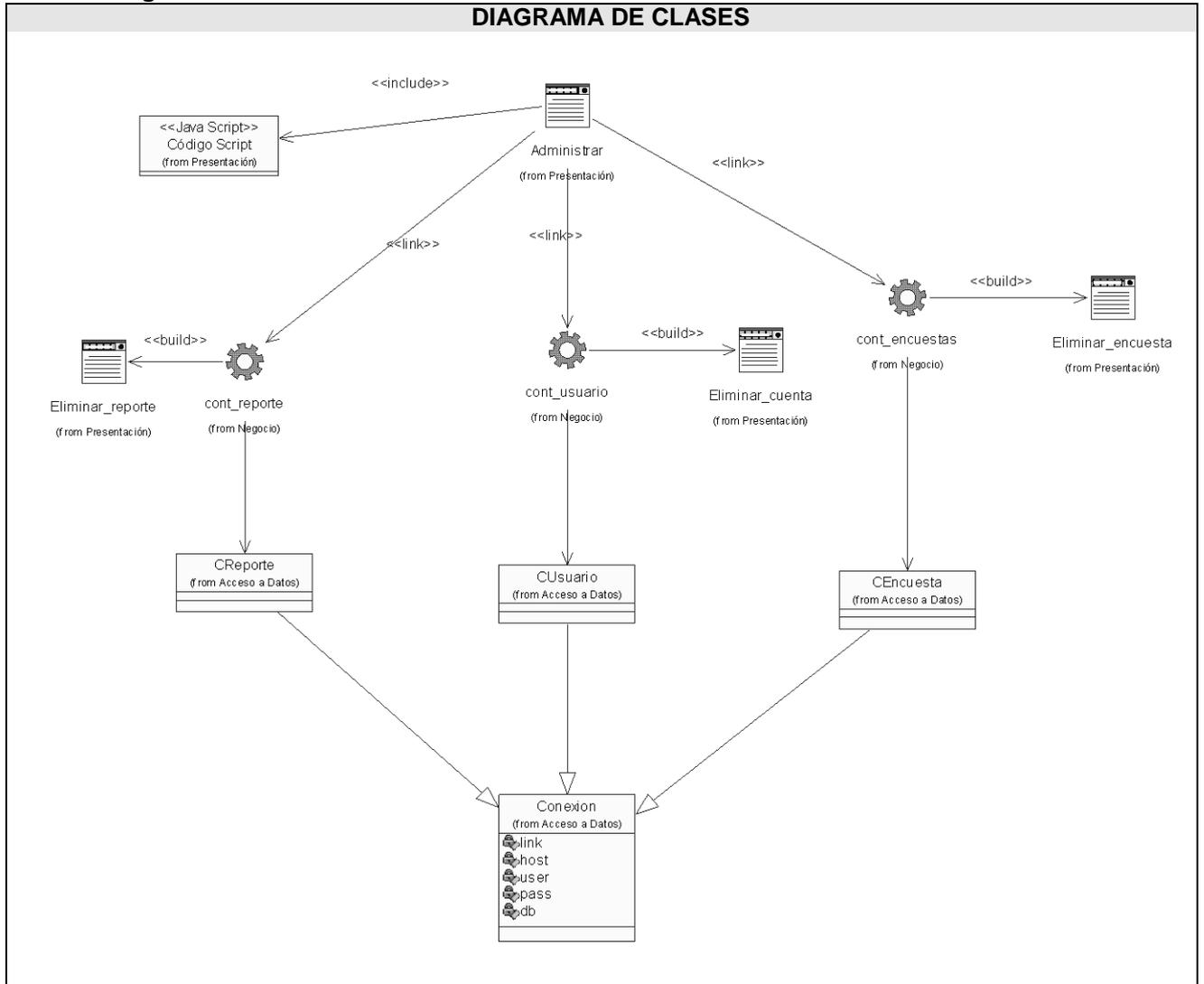


Figura 3. 17 Diagrama de clases del diseño CU: Eliminar Elementos Innecesarios.

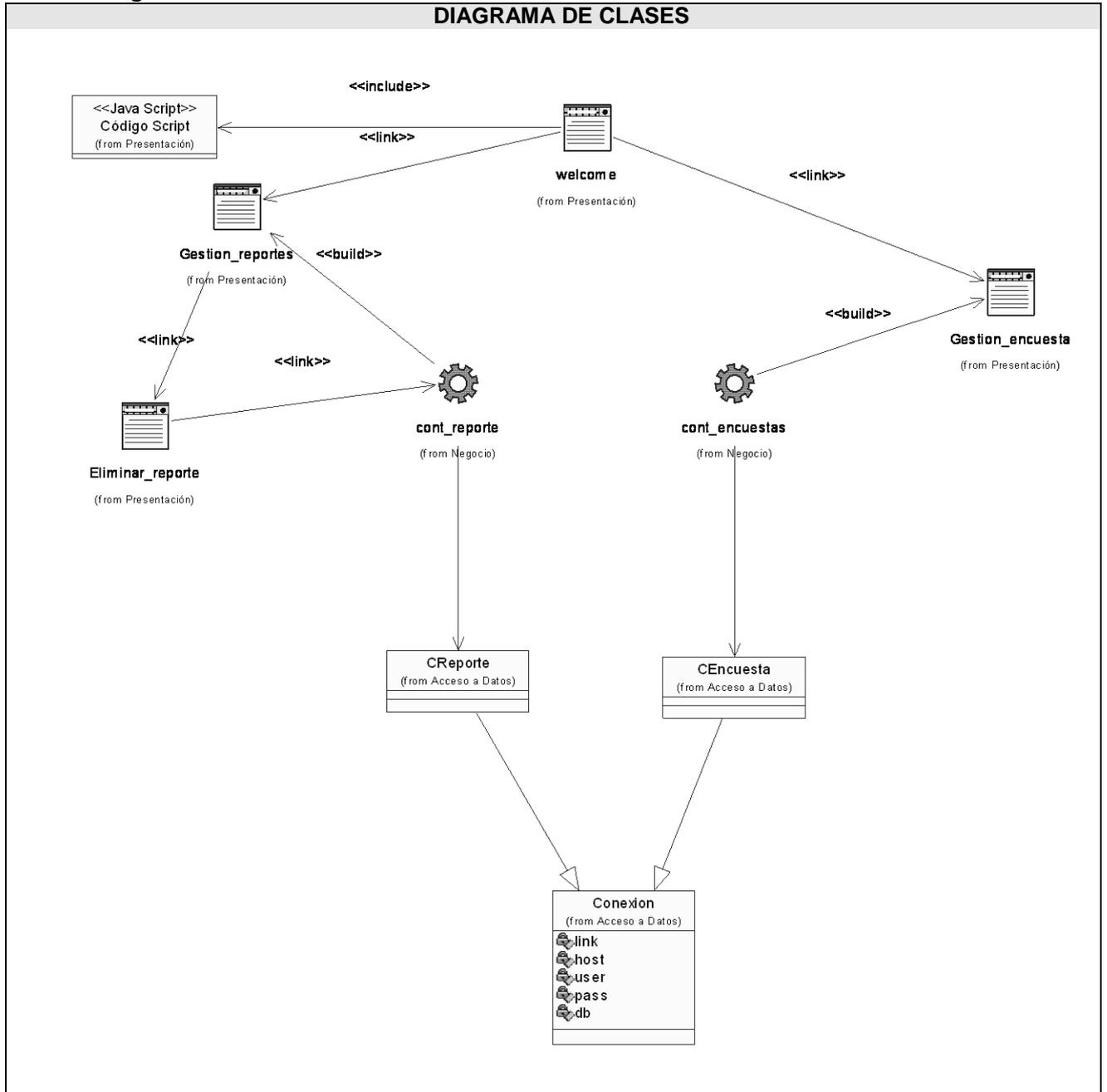


Figura 3. 18 Diagrama de clases del diseño CU: Mostrar Usuarios Registrados.

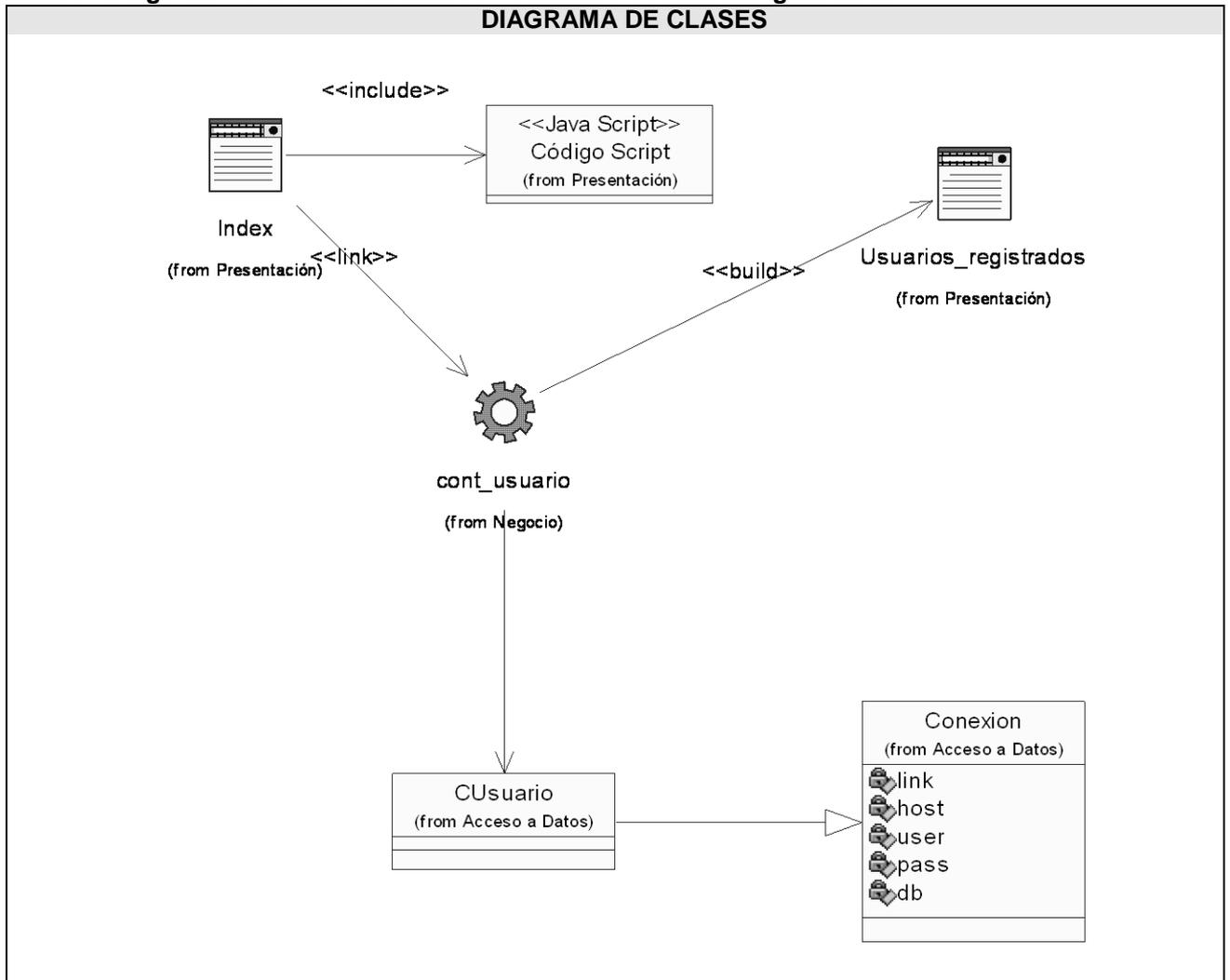


Figura 3. 19 Diagrama de clases del diseño CU: Solicitar Cuenta.

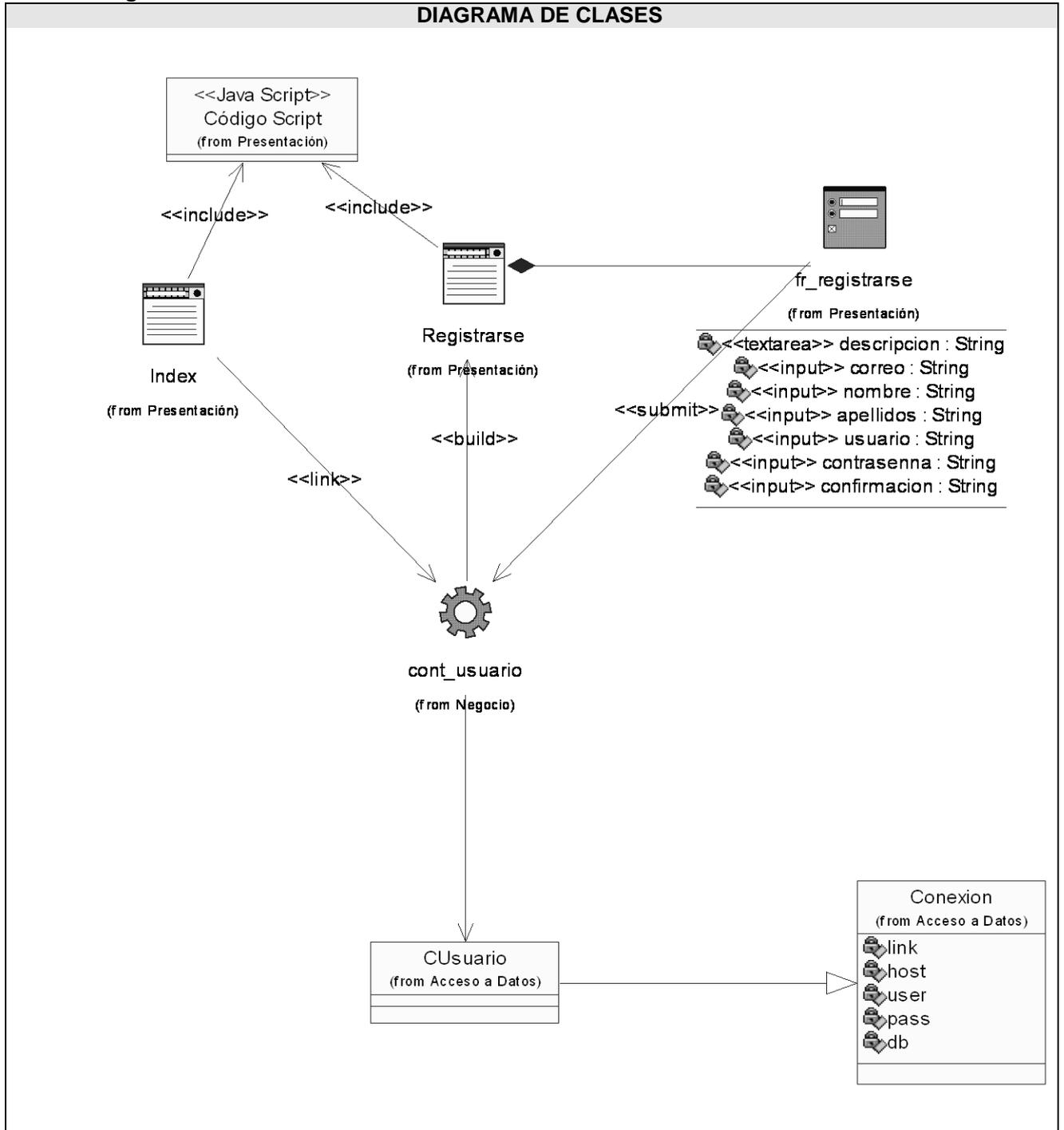


Figura 3. 20 Diagrama de clases del diseño CU: Configurar Reporte.

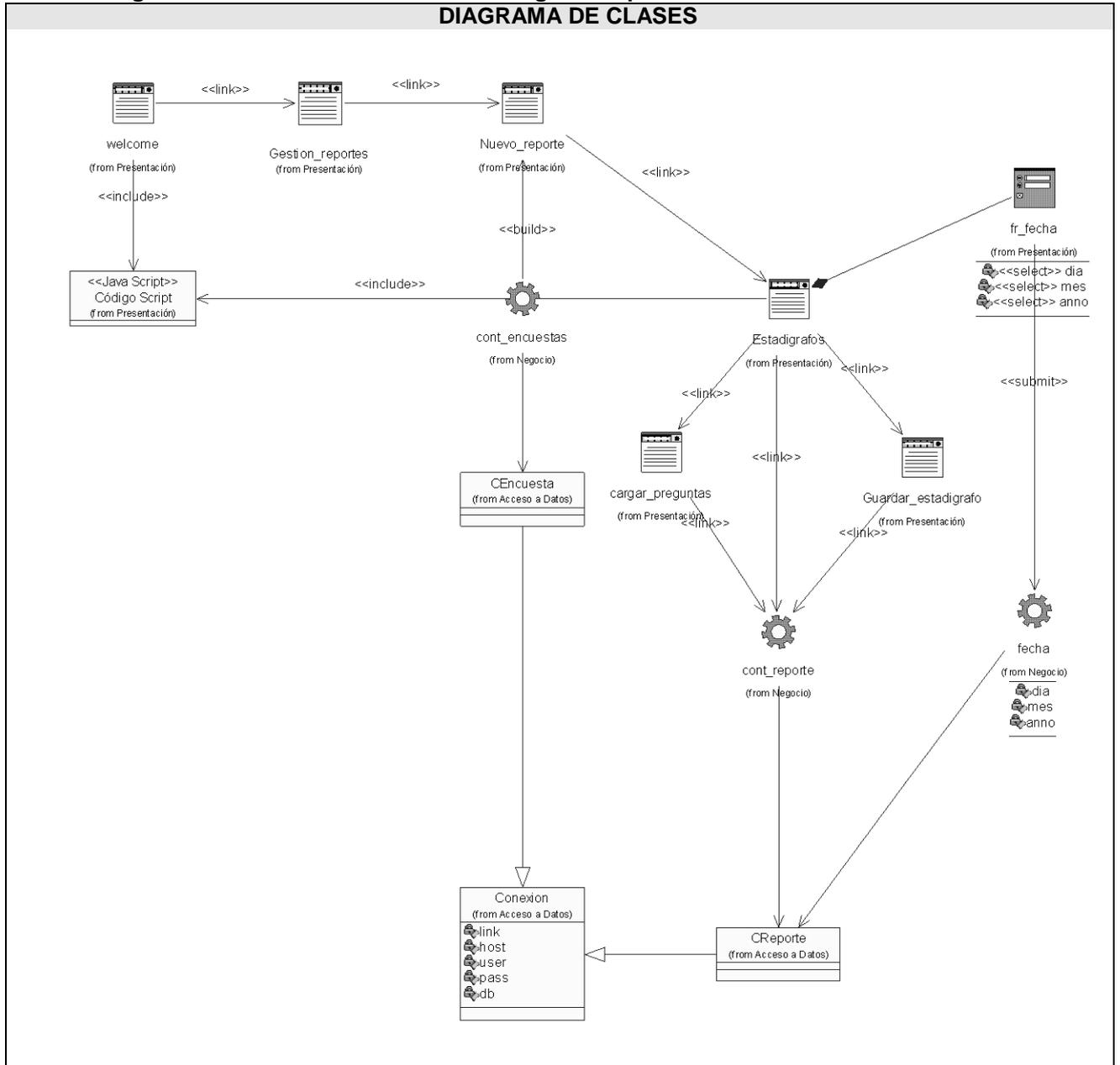


Figura 3. 21 Diagrama de clases del diseño CU: Enviar Reporte.

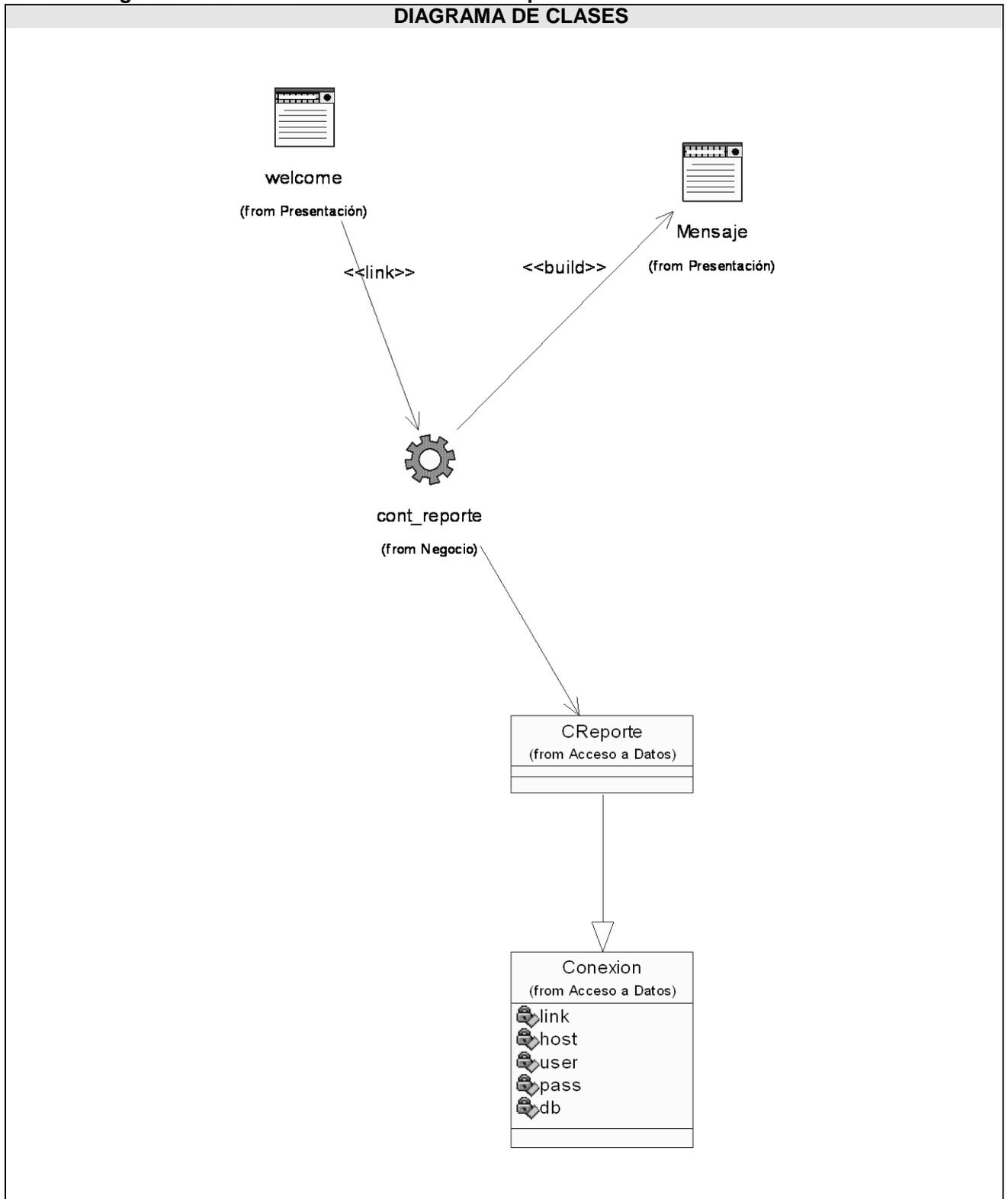
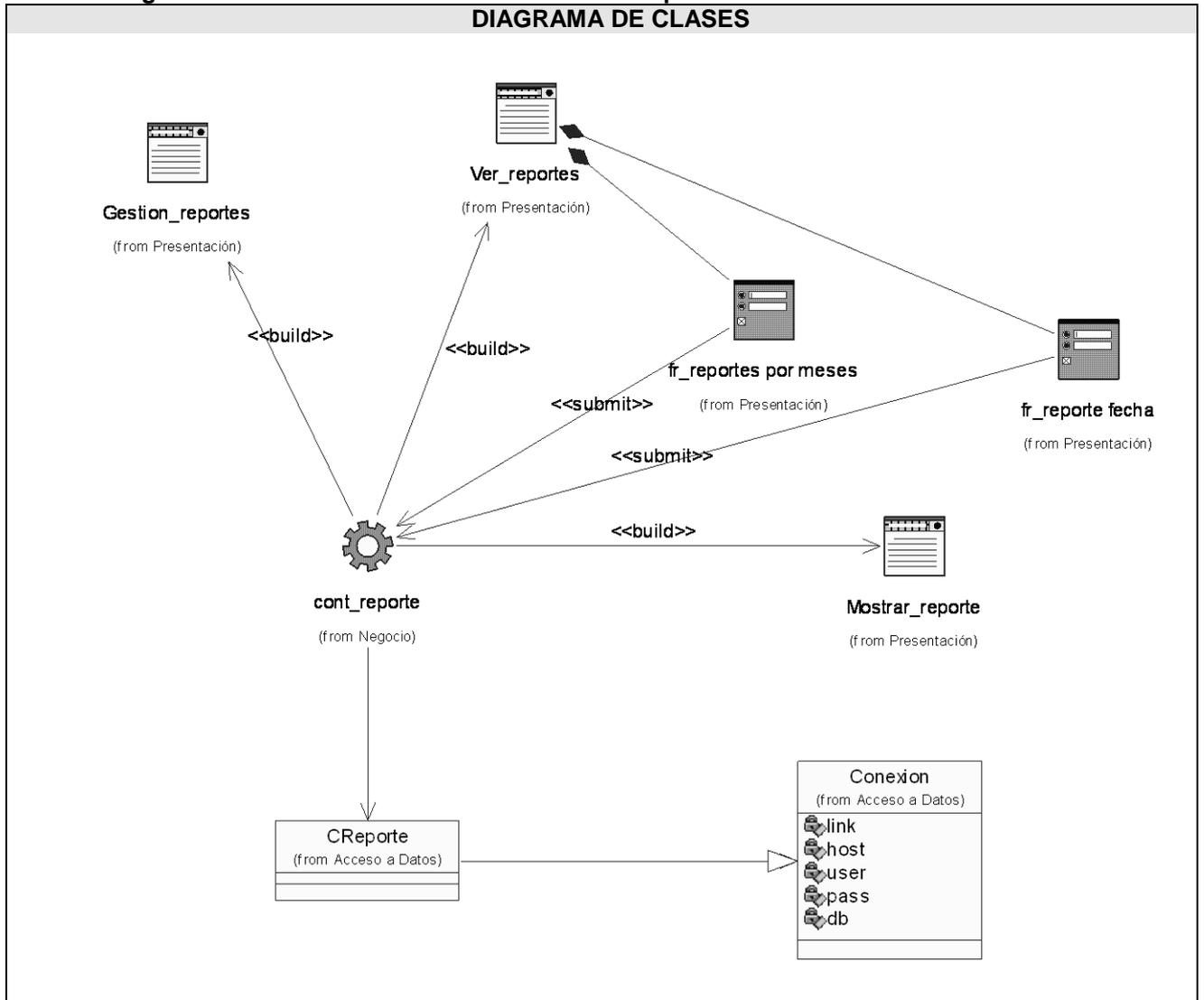


Figura 3. 22 Diagrama de clases del diseño CU: Consultar Reporte.



3.3.4 Descripción de las clases.

A continuación se realiza una descripción de las clases con las que cuenta el sistema y las principales funciones que poseen.

Tabla 3. 1 Clase cont_usuario.

Nombre: cont_usuario	
Tipo de clase: Controladora	
Atributo	Tipo
usuario	
Para cada responsabilidad:	
Nombre	Descripción

function cont_usuario()	Es el constructor de la clase
function solicitud (\$anombres, \$aapellidos, \$ausuario, \$apassword, \$acorreo, \$adescripcion)	Realiza el llamado a funcion solicitud_cuenta que permite insertar una solicitud de cuenta en la base de datos.
function solicitudes()	Realiza el llamado a funcion mostrar_solicitudes() que permite mostrar las solicitudes de cuenta que hay en la base de datos.
function eliminar_solicitud(\$usuario)	Realiza el llamado a funcion eliminar_solicitud(\$usuario) que permite eliminar la solicitud de un usuario de la base de datos
function aceptar_solicitud(\$usuario)	Realiza el llamado a funcion crear_cuenta(\$usuario) que permite insertar una cuenta en la base de datos.
function usuarios_registrados()	Realiza el llamado a funcion usuarios_registrados() que permite mostrar los usuarios que se encuentran registrados en la base de datos con cuentas locales.
function posibles_administradores()	Permite devolver una lista con los usuarios que pueden ser administradores
function nuevo_administrador(\$administrador)	Realiza el llamado a funcion insertar_administrador(\$administrador) que permite insertar un usuario determinado como administrador del sistema.
function existe_usuario(\$usuario)	Verifica la existencia de un usuario.
function actualizar_cuenta_usuario(\$id_usua,\$fech)	Actualiza la cuneta de usuario con la última fecha de autenticación.
function mostrar_fecha(\$id_usuario)	Muestra la fecha en la que el usuario se autenticó por última vez
function eliminar_usuario(\$id_usuario)	Realiza un llamado a funcion eliminar_cuenta(\$id_usuario) que permite eliminar la cuenta del usuario de la base de datos.

Tabla 3. 2 Clase cont_reporte.

Nombre: cont_reporte	
Tipo de clase: Controladora	
Atributo	Tipo
c_reporte	
Para cada responsabilidad:	
Nombre	Descripción
function cont_reporte()	Es el constructor de la clase
function nuevo_reporte(\$fecha,\$usser)	Realiza el llamado a funcion crear_reporte(\$fecha, \$usser) que permite realizar la solicitud de un reporte.
function reporteActual()	Devuelve el id del reporte sobre el que se está trabajando.

function solicitud(\$id,\$fecha)	Realiza el llamado a funcion guardar_solicitud(\$id,\$fecha) que guarda la configuración del reporte.
function nuevoTipoEstadigrafo(\$nombre)	Realiza el llamado a funcion insertar_tipoEstadigrafo(\$nombre) que inserta el tipo de estadigrafo seleccionado
function TipoEstadigrafoActual()	Devuelve el id de estadigrafo que se está aplicando a las preguntas.
function nuevoEstadigrafo(\$id)	Realiza el llamado a funcion insertar_Estadigrafo(\$id) que inserta el tipo de estadigrafo con las preguntas a las que se le aplica.
function nuevoReporteEstadigrafo(\$id_report, \$id_estadig)	Realiza el llamado a funcion insertar_Reporte_Estadigrafo(\$id_report,\$id_estadig) que inserta del reporte señalando, los estadígrafos que se desea aplicar.
function EstadigrafoActual()	Devuelve el estadigrafo sobre el que se está trabajando.
function nuevoEstadigrafoPregunta(\$id_estadig, \$id_pregunt)	Realiza el llamado a funcion insertar_pregunta_estadigrafo(\$id_estadig, \$id_pregunt) que inserta una pregunta para un estadígrafo señalado.
function fecha_reporte(\$usuario)	Devuelve las fechas en que fueron generados los reportes que posee el usuario.
function Cargar_id_reportes_nuevos(\$usuario,\$fecha)	Muestra el id de los reportes nuevos que posee el usuario en la fecha señalada.
function mostrar_reportes_nuevos(\$usuario,\$fecha)	Muestra los reportes nuevos que posee el usuario en la fecha señalada.
function reportes_fecha(\$usuario,\$fecha_ini,\$fecha_f)	Muestra los reportes que posee el usuario en un rango de tiempo.
function path_reporte(\$reporte)	Devuelve la dirección en que se ha guardado el reporte.
function nombre_encuesta(\$id_encuesta)	Devuelve el nombre de la encuesta de la que se obtiene el reporte
function generar_reporte(\$id_report, \$user, \$dir)	Genera un reporte en formato PDF.

Tabla 3. 3 Clase cont_encuestas

Nombre: cont_encuestas	
Tipo de clase: Controladora	
Atributo	Tipo
c_encuesta	
Para cada responsabilidad:	
Nombre	Descripción
function cont_encuestas()	Es el constructor de la clase
function mostrarEncuestas(\$usuario)	Realiza el llamado a funcion mostrar_encuestas(\$usuario) que permite mostrar

	las encuestas que posee un determinado usuario.
function mostrarTodasEncuestas()	Realiza el llamado a function mostrar_todas_encuestas() que permite mostrar todas las encuestas de la base de datos.
function eliminar(\$id)	Realiza el llamado a function eliminar_encuesta(\$id) que permite eliminar encuestas de la base de datos.
function mostrar_numericas(\$id)	Realiza el llamado a function mostrar_preguntas_numericas(\$id) que permite mostrar las preguntas numéricas de una determinada encuesta
function mostrar_categoricas(\$id)	Realiza el llamado a function mostrar_preguntas_categoricas(\$id) que permite mostrar las preguntas categóricas de una determinada encuesta

Tabla 3. 4 Clase Conexion

Nombre: Conexion	
Tipo de clase: Acceso a datos	
Atributo	Tipo
link	Id de la conexión
host	Servidor al que se realiza la conexión(localhost)
user	Usuario de la BD
pass	Contraseña del usuario.
db	BD
Para cada responsabilidad:	
Nombre	Descripción
function Conexion()	Crear una conexión a la BD
function selectBD(\$bd)	Selecciona la BD
function query(\$query)	Ejecuta una consulta a la BD
function queryAsArray(\$_query)	Devuelve un arreglo de la consulta ejecutada
function Close()	Cierra la conexión a la BD

Tabla 3. 5 Clase CEncuesta

Nombre: CEncuesta extends Conexion	
Tipo de clase: Acceso a datos	
Atributo	Tipo
Para cada responsabilidad:	
Nombre	Descripción
function CEncuesta()	Es el constructor de la clase
function encuestas_caducadas()	Muestra las encuestas que llevan más de 6 meses sin ser publicadas
function eliminar_encuesta(\$id_encuesta)	Eliminar encuestas de la base de datos.
function mostrar_encuestas(\$_usuario)	Mostrar las encuestas que posee un determinado usuario.
function mostrar_todas_encuestas()	Mostrar todas las encuestas de la base de datos.

function mostrar_preguntas_numericas(\$id_encuesta)	Mostrar las preguntas numéricas de una determinada encuesta
function mostrar_preguntas_categoricas(\$id_encuesta)	Mostrar las preguntas categóricas de una determinada encuesta

Tabla 3. 6 Clase CUsuario.

Nombre: CUsuario extends Conexion	
Tipo de clase: Acceso a datos	
Atributo	Tipo
Para cada responsabilidad:	
Nombre	Descripción
function CUsuario()	Es el constructor de la clase
function solicitud_cuenta (\$_nombres, \$_apellidos, \$_usuario, \$_password, \$_correo, \$_descripcion)	Insertar una solicitud de cuenta en la base de datos
function crear_cuenta(\$_usuario)	Insertar una cuenta en la base de datos
function eliminar_cuenta()	Elimina una cuenta de la base de datos
function eliminar_solicitud(\$_usuario)	Elimina la solicitud de un usuario de la base de datos
function mostrar_solicitudes()	Muestra las solicitudes de cuenta que hay en la base de datos
function usuarios_registrados()	Muestra los usuarios que se encuentran registrados en la base de datos con cuentas locales.
function usuarios()	Seleccionar todos los usuarios
function administradores()	Seleccionar todos los administradores
function insertar_administrador(\$admon)	Inserta un usuario determinado como administrador del sistema.
function actualizar_cuenta(\$id_usuario,\$fecha)	Actualiza la cuenta de usuario con la última fecha en la que éste se ha autenticado.
function fecha_utilizada(\$id)	Devuelve la fecha en la que el usuario se autenticó por última vez.

Tabla 3. 7 Clase CReporte.

Nombre: CReporte extends Conexion	
Tipo de clase: Acceso a datos	
Atributo	Tipo
Para cada responsabilidad:	
Nombre	Descripción
function CReporte()	Es el constructor de la clase
function crear_reporte(\$fecha,\$usuario)	Permite realizar la solicitud de un reporte.
function reporte_actual()	Devuelve el reporte en el que está trabajando.
function guardar_solicitud(\$id_reporte,\$fecha)	Guarda la configuración del reporte.
function insertar_tipoEstadigrafo(\$nombre)	Inserta el tipo de estadígrafo seleccionado.
function tipo_actual()	Muestra el tipo de estadígrafo en el que se están insertando las preguntas.
function insertar_Estadigrafo(\$id_tipo)	Insertar el tipo de estadígrafo con las

	preguntas a las que se le aplica.
function insertar_Reporte_Estadigrafo(\$id_reporte, \$id_estadi)	Insertar del reporte señalando, los estadígrafos que se desea aplicar
function estadigrafo_actual()	Devuelve el estadígrafo actual
function insertar_pregunta_estadigrafo(\$id_esta, \$id_preg)	Inserta una pregunta para un estadígrafo señalado.
function fecha_generado(\$usuario)	Selecciona las fechas en que se generan los reportes del usuario
function aviso_reportes_nuevo(\$usuario,\$fecha)	Muestra aviso de nuevo reporte
function reportes_fecha(\$usuario,\$fecha_inicio, \$fecha_fin)	Muestra los reportes que posee el usuario en un determinado rango de tiempo.
function encuesta(\$id)	Devuelve el nombre de la encuesta a la que se le aplica el estadígrafo.
function nombres_estadigrafos(\$id_reporte)	Devuelve los estadígrafos que se han aplicado en el reporte
function nuevo_reporte_generado(\$id_reporte, \$usuario, \$path)	Inserta un reporte nuevo sin mostrar
function actualizar_reporte_generado(\$id_reporte)	Una vez que se haya mostrado el reporte lo actualiza
function path(\$id_reporte)	Devuelve la dirección en la que se ha generado el reporte.

Tabla 3. 8 Clases Interfaz de Usuario

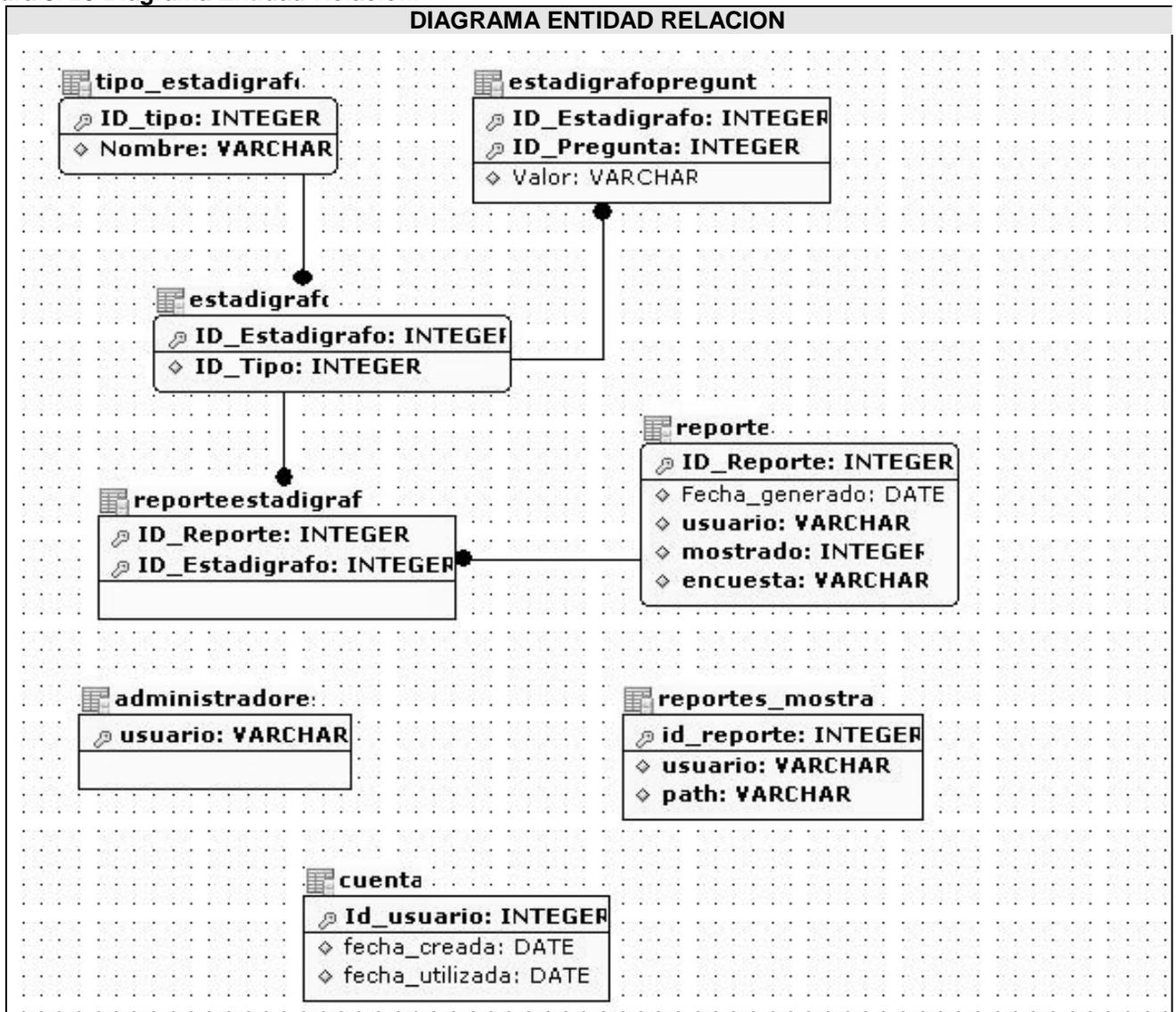
Nombre	Descripción
Index.php	Página de inicio de la aplicación. Permite la autenticación de los usuarios.
welcome.php	Página que se visualiza una vez que el usuario se ha autenticado.
Administrar.php	Página que posee las funcionalidades que puede ejecutar el administrador del sistema.
conectarse.php	Permite realizar un conjunto de funciones relacionadas con la autenticación de usuarios.
Editar_Perfil.php	Página que permite cambiar los datos de la cuenta de usuario.
Eliminar_cuenta.php	Muestra una lista con las cuentas de usuario y la fecha del último acceso al sistema y permite seleccionar cual de estas se desea eliminar.
Eliminar_encuesta.php	Muestra una lista de las encuestas que hay en el sistema y permite seleccionar cual se desea eliminar.
Eliminar_reporte.php	Permite eliminar reportes del sistema.
Eliminar_solicitud.php	Permite eliminar una solicitud de cuenta de usuario
Gestion_Encuestas.php	Muestra las encuestas del usuario y permite seleccionar cual desea eliminar.
Gestion_reportes.php	Página que posee las funcionalidades que puede realizar un encuestador sobre sus reportes.

Registrar_cuenta.php	Página que permite al administrador crear una cuenta a partir de una solicitud determinada.
Registrarse.php	Página que permite a los usuarios anónimos realizar la solicitud de una cuenta local.
Usuarios_registrados.php	Página que muestra una lista de los usuarios registrados en el sistema.
Ver_reportes.php	Página que permite la consulta de los reportes en un rango de tiempo determinado, además muestra los reportes que posee el usuario pero que no han sido consultados por éste.
Nuevo_reporte.php	Muestra las encuestas que posee un usuario y permite seleccionar de cual de éstas desea el reporte.
Mensaje.php	Muestra un mensaje indicando al usuario que posee nuevos reportes.
Guardar_estadigrafo.php	Permite guardar los datos de la configuración del reporte.
Estadígrafos.php	Permite seleccionar todos los datos para la configuración del reporte una vez que se haya seleccionado la encuesta a la que se le va a aplicar el mismo.
Crear_Reporte.php	Permite crear un nuevo reporte a partir de la configuración realizada por el usuario.
cargar_preguntas.php	Carga las preguntas a las que se les puede aplicar determinado estadígrafo.
Nuevo_administrador.php	Muestra una lista de los usuarios del sistema que no son administradores y permite seleccionar a cual de éstos se les desea otorgar los privilegios de administración.
PDF.php	Página que permite generar el reporte en formato PDF.

3.3.5 Diseño de la BD

A continuación se describen las tablas que se adicionaron a la base de datos con que contaba el Sistema de Análisis Estadístico de Datos.

Figura 3. 23 Diagrama Entidad-Relación.



3.3.5.1 Descripción de las tablas.

Tabla 3. 9 Reporte

Nombre: reporte		
Descripción: Esta tabla almacena los datos de los reportes configurados por cada usuario.		
Atributo	Tipo	Descripción
ID_Reporte	Integer	Llave primaria, identificador del reporte.
Fecha_generado	Date	Fecha en la que se genera el reporte.
Usuario	Varchar	Usuario que realiza la configuración del reporte.
Mostrado	Tinyint	Indica si el reporte ha sido consultado o no.

Tabla 3. 10 Estadígrafo

Nombre: estadígrafo		
Descripción: Esta tabla almacena todos los estadígrafos.		
Atributo	Tipo	Descripción
ID_Estadígrafo	Integer	Llave primaria, identificador del estadígrafo.
ID_Tipo	Integer	Llave foránea, identificador del tipo de estadígrafo

Tabla 3. 11 Tipo _ estadígrafo

Nombre: tipo _ estadígrafo		
Descripción: Esta tabla almacena los tipos de estadígrafos.		
Atributo	Tipo	Descripción
ID_Tipo	Integer	Llave primaria, identificador del tipo de estadígrafo.
Nombre	varchar	Nombre del tipo de estadígrafo.

Tabla 3. 12 Estadígrafopregunta.

Nombre: estadígrafopregunta		
Descripción: Esta tabla almacena las preguntas a las que se les puede aplicar cada estadígrafo.		
Atributo	Tipo	Descripción
ID_Estadígrafo	Integer	Forma parte de la llave primaria,
ID_Pregunta	Integer	Forma parte de la llave primaria
valor	float	Valor que posee el estadígrafo aplicado a la pregunta.

Tabla 3. 13 Reporteestadígrafo

Nombre: reporteestadígrafo		
Descripción: Esta tabla almacena los estadígrafos que hay en cada reporte.		
Atributo	Tipo	Descripción
ID_Reporte	Integer	Forma parte de la llave primaria
ID_Estadígrafo	Integer	Forma parte de la llave primaria

Tabla 3. 14 Reporte _ mostrar.

Nombre: reporte _ mostrar		
Descripción: Esta tabla almacena los reportes de cada usuario.		
Atributo	Tipo	Descripción
ID_Reporte	Integer	Llave primaria, identificador del reporte a mostrar.
Usuario	Varchar	

Tabla 3. 15 Administradores.

Nombre: administradores		
Descripción: Esta tabla almacena los administradores del sistema		
Atributo	Tipo	Descripción
Usuario	Varchar	Almacena el nombre de los usuarios que son administradores del sistema.

Tabla 3. 16 Cuenta.

Nombre: cuenta		
Descripción: Esta tabla almacena las cuentas.		
Atributo	Tipo	Descripción
Id_usuario	Integer	Es la llave primaria, identifica al usuario que posee la

		cuenta.
fecha_creada	Date	Almacena la fecha en que fue creada la cuenta.
fecha_utilizada	Date	Almacena la fecha en que se utilizó la cuenta por última vez.

Tabla 3. 17 Solicitud_cuenta.

Nombre: solicitud_cuenta		
Descripción: Esta tabla almacena las solicitudes de cuentas locales.		
Atributo	Tipo	Descripción
usuario	Varchar	Es la llave primaria, identifica la solicitud
nombre	Varchar	Almacena el nombre del usuario que solicita la cuenta.
apellido	Varchar	Almacena el apellido del usuario que solicita la cuenta.
correo	Varchar	Almacena la dirección de correo electrónico del usuario que solicita la cuenta.
descripcion	Varchar	Almacena la descripción realizada por el usuario que solicita la cuenta.
password	Varchar	Almacena la contraseña del usuario que solicita la cuenta.

3.3.6 Principios de Diseño.

El diseño debe basarse en el usuario, y en este caso se está hablando de cualquier tipo de persona que desee llenar una encuesta, aunque no tenga gran preparación en cuestiones informáticas. Para ello, este sistema utiliza ciertos principios generales que garantizan la usabilidad en los diseños para aplicaciones Web.

1. Principio de uso equiparable: donde las características de privacidad, garantía y seguridad estén igualmente disponibles para todos los usuarios, y que el diseño sea atractivo para los mismos.
2. Principio de la flexibilidad: donde se ofrezcan posibilidades de elección en los métodos de uso, que facilite al usuario la exactitud y precisión, y se adapte al paso o ritmo del usuario.
3. Principio de la Información perceptible: donde se usen diferentes modos para presentar de manera redundante la información esencial (verbal y tablas), se proporcione contraste suficiente entre la información esencial y sus alrededores.
4. Principio de tolerancia al error: donde se dispongan los elementos para minimizar los riesgos y errores, por ejemplo utilizando elementos comunes; y los elementos peligrosos eliminados, aislados o tapados, que se proporcionen advertencias sobre peligros y errores. Hay que

posibilitar el descubrimiento interactivo y el aprendizaje ensayo-error, y posibilitar la reversibilidad y la recuperabilidad de las acciones.

5. Principio de esfuerzo de acceso y uso: que minimicen las acciones repetitivas, y que proporcione una línea de visión clara hacia los elementos importantes tanto para un usuario sentado como de pie.

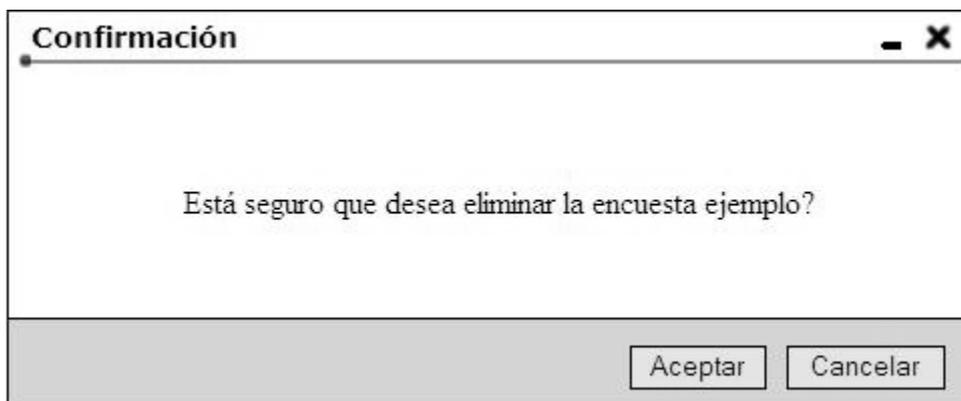
3.3.7 Tratamiento de errores

Hay casos que incluye la forma de solucionar el error, como es el caso de la autenticación de los usuarios.



usuario: **Acceso Denegado!!**
contraseña:

Algunos errores serán generados para evitar la ejecución de la página en vano. Se utilizan errores en forma de mensajes de texto en la misma página donde se ejecutó la acción, de forma que el usuario pueda corregir más fácilmente y continuar. Y se utilizan mensajes de confirmación, para acciones que son irreversibles como es el caso de las eliminaciones.



Confirmación - X

Está seguro que desea eliminar la encuesta ejemplo?

3.3.8 Seguridad

Debido a la importancia y sensibilidad de los datos que se manejarán en la aplicación se hace necesario que se garantice la protección y seguridad de la información para el óptimo funcionamiento

del sistema. La seguridad en el sitio estará implementada a través del uso de sesiones para restringir el acceso hacia los datos que no se tenga permiso.

Las contraseñas de los usuarios se guardan en la base de datos encriptadas, utilizando la función md5, una de las características más importantes de dicha función es su irreversibilidad, una vez que se encripte una contraseña no podrá ser descifrada, lo que garantiza la seguridad de los datos de los usuarios.

3.3.9 Concepción de la ayuda.

La concepción del diseño facilita la navegación y contribuye a disminuir la necesidad de una ayuda para el usuario, no obstante el sistema orienta al usuario en las páginas de bienvenida a los módulos Configuración y Análisis Estadístico, donde explica las principales opciones existentes dentro del módulo en el que se encuentre, las cuales son accesibles desde cualquier otra página.

3.4 Conclusiones

En este capítulo se mostró el análisis y diseño del sistema a partir de la realización de los diagramas de clases correspondientes tanto en el análisis como en el diseño y además se elaboraron los diagramas de interacción correspondientes a cada escenario de los casos de uso definidos en el capítulo anterior. También se elaboró el diagrama entidad relación que muestra las tablas que posee la base de datos del Sistema de Análisis Estadístico de Datos para los nuevos módulos de administración y reportes.

Capítulo 4: Implementación y Pruebas.

4.1 Introducción

En este capítulo se construyen los artefactos del modelo de implementación, para esto se elabora el diagrama de despliegue del sistema así como la estructura del modelo de implementación y los diagramas de componentes que la conforman.

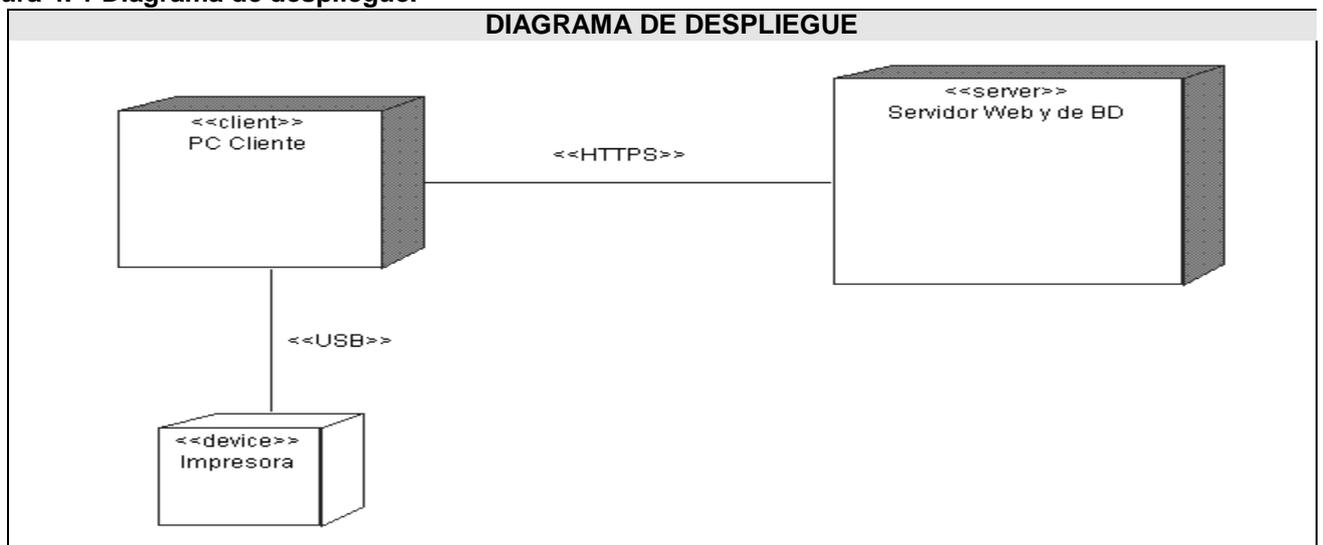
4.2 Implementación.

4.2.1 Diagrama de despliegue.

El diagrama de despliegue muestra la configuración de los diferentes dispositivos de hardware que participan en la ejecución del sistema.

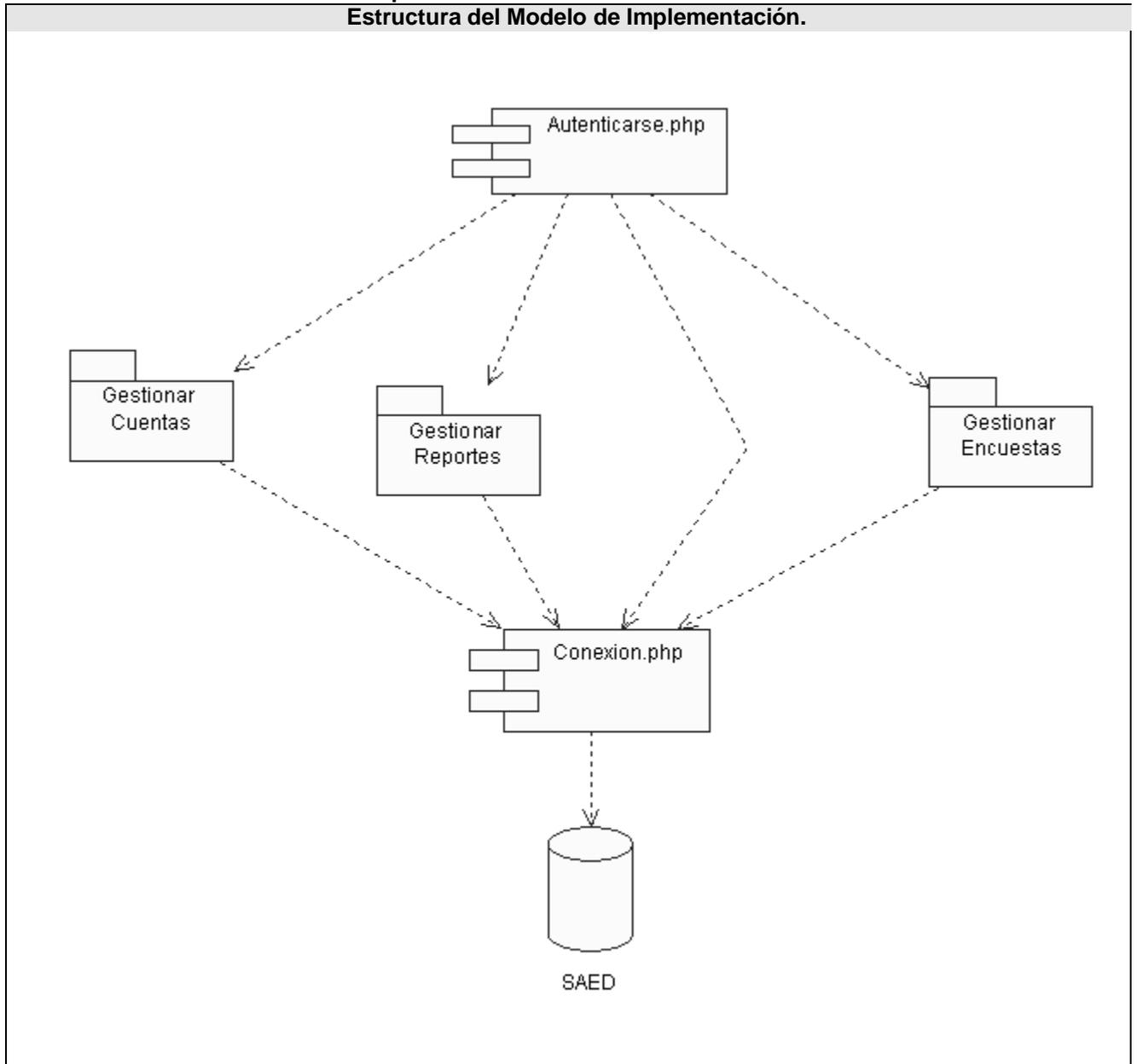
El diagrama de despliegue que se ha elaborado cuenta con un nodo de procesamiento que contendrá al Servidor Web y Servidor de datos. El Cliente solicita los servicios vía HTTPS para establecer una conexión segura y tendrá conectada, una impresora en el caso de que los usuarios necesiten imprimir algún recurso.

Figura 4. 1 Diagrama de despliegue.



4.2.2 Estructura del modelo de Implementación

Figura 4. 2 Estructura del Modelo de Implementación.



4.2.3 Diagramas de componentes

Figura 4. 3 Diagrama de Componentes: Gestionar Cuenta.

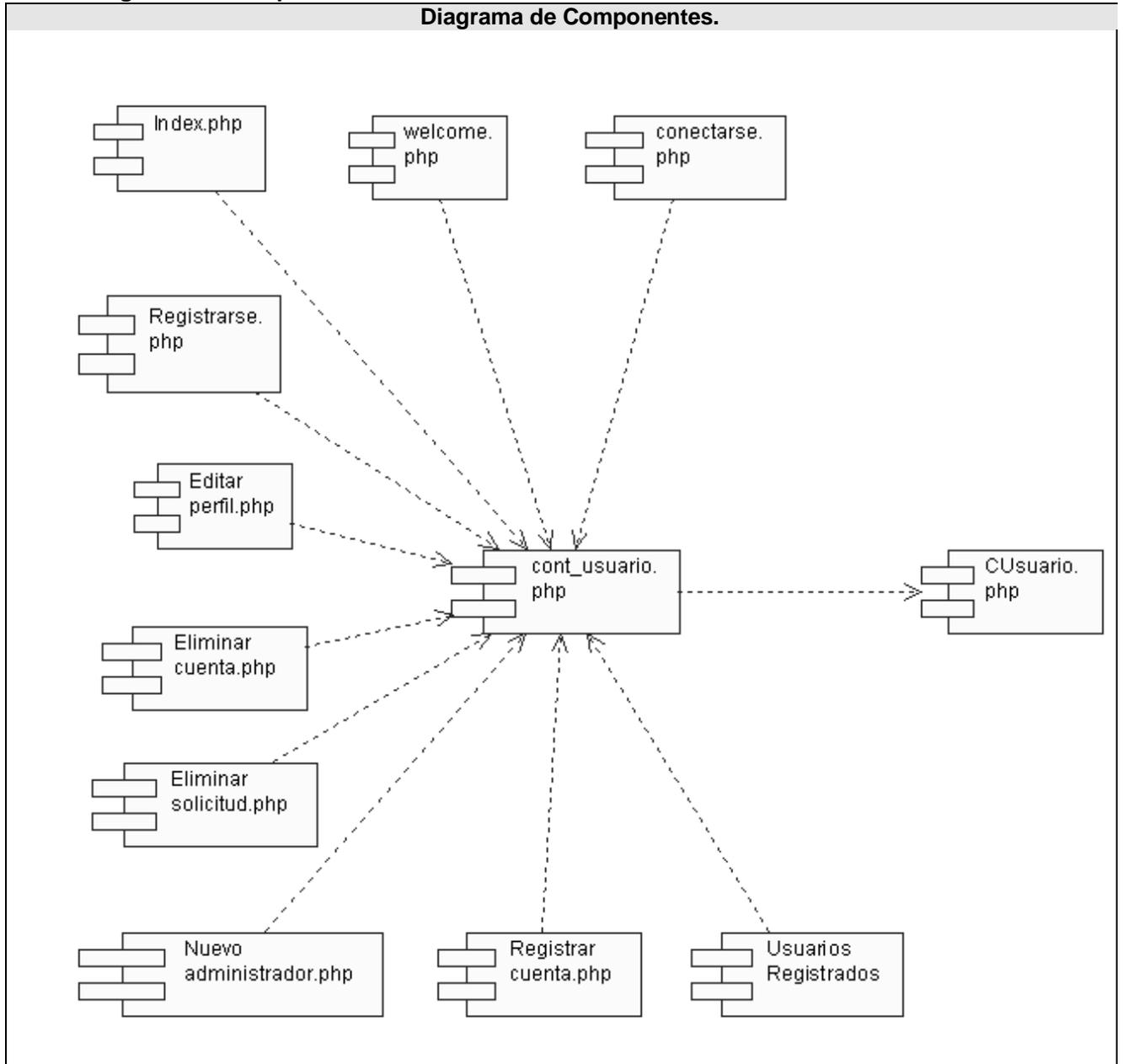


Figura 4. 4 Diagrama de Componentes: Gestionar Reporte.

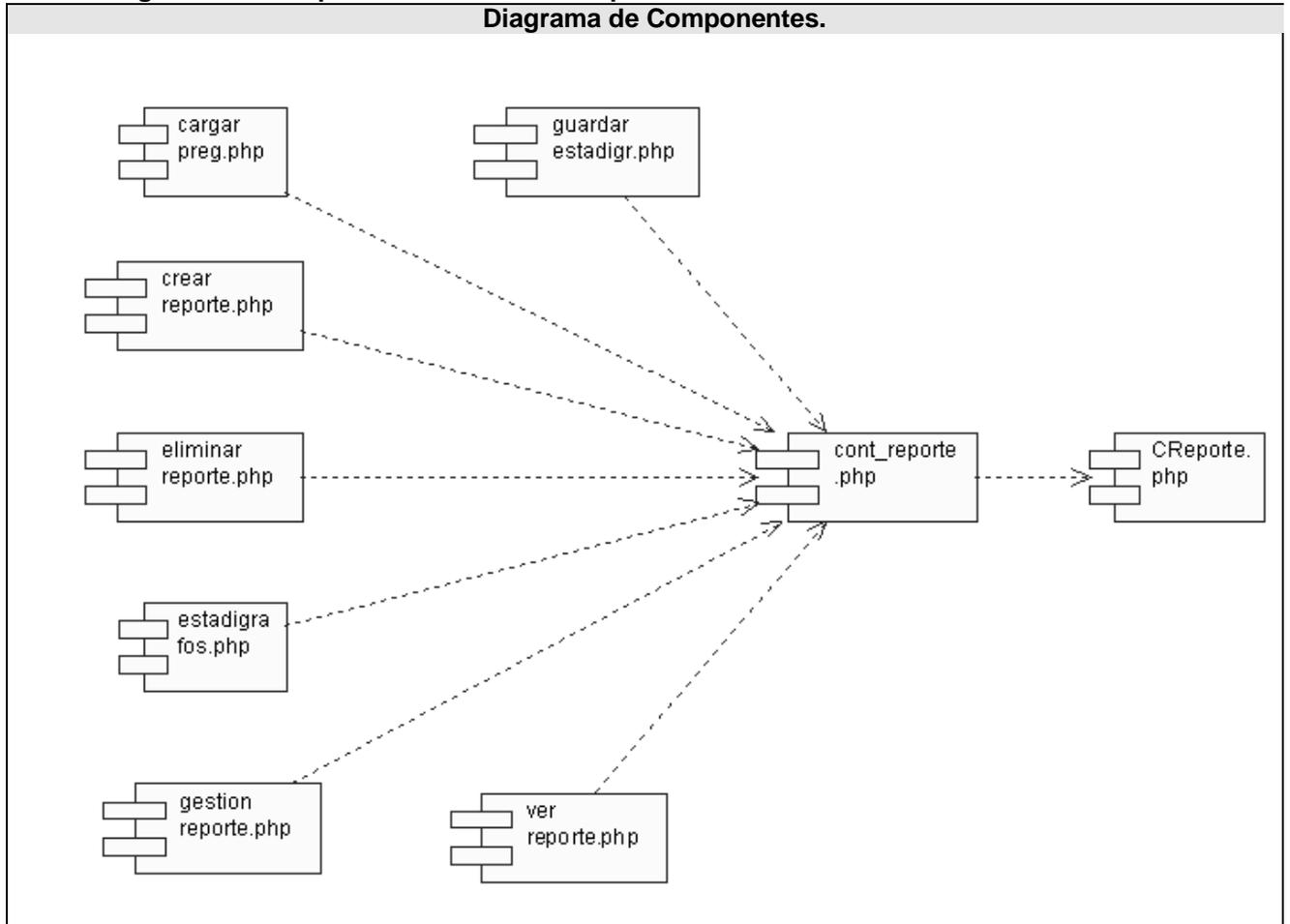
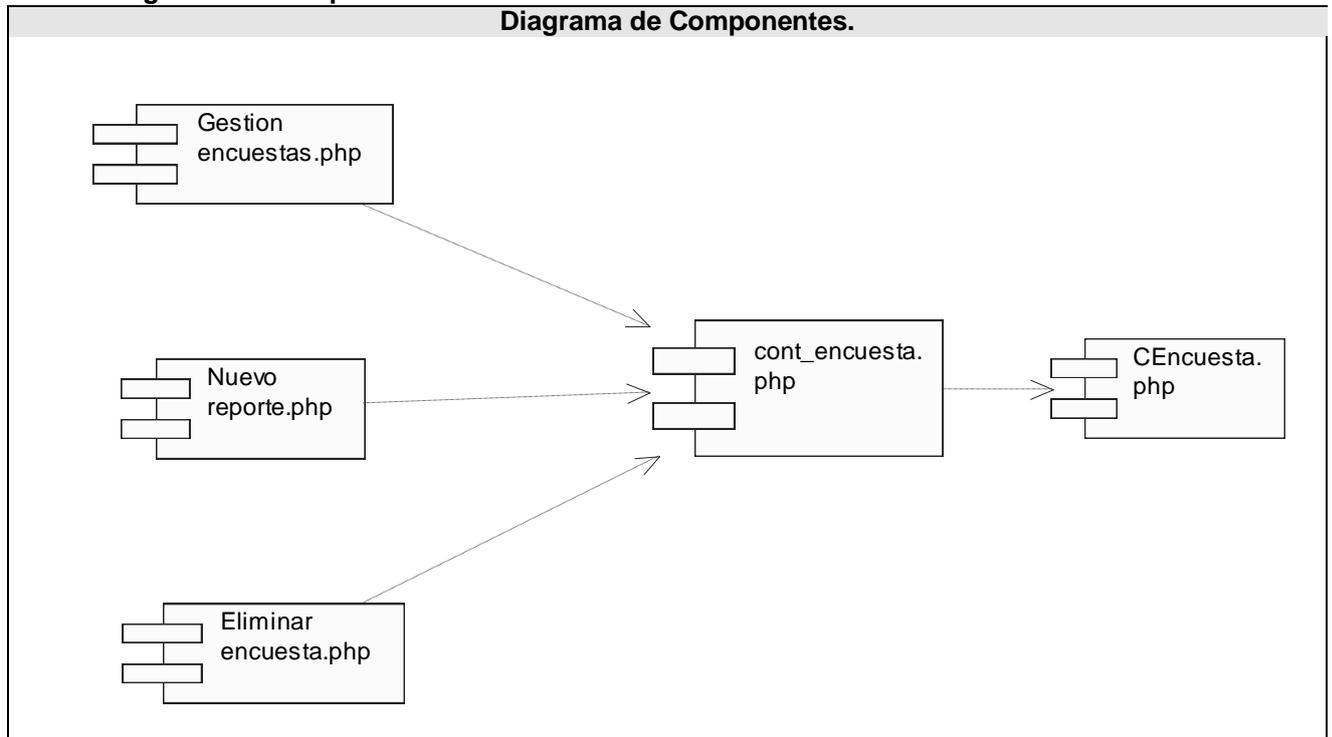


Figura 4. 5 Diagrama de Componentes: Gestionar Encuesta.



4.3 Modelo de Prueba.

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

4.3.1 Pruebas de caja negra

Estas pruebas se realizan sobre la interfaz del software y con ellas se pretende demostrar no sólo que las funciones del software son operativas sino también que la entrada se acepta de forma adecuada y que se produce una salida correcta; además que la integridad de la información externa se mantiene.

Las pruebas de caja negra se centran principalmente en los requisitos funcionales del software. Estas pruebas permiten obtener un conjunto de condiciones de entrada que ejerciten completamente todos los requisitos funcionales de un programa. En ellas se ignora la estructura de control, concentrándose en los requisitos funcionales del sistema y ejercitándolos.

Estas pruebas son de vital importancia para el software pues mediante ellas se pueden encontrar funciones incorrectas o ausentes, errores de interfaz, errores en estructuras de datos o en accesos a las Bases de Datos externas, errores de rendimiento así como errores de inicialización y terminación.

4.3.2 Casos de prueba

Tabla 4. 1 Caso de Prueba CU: Autenticar Usuario.

Condición de Entrada	Casos Válidos	Casos no Válidos
Usuario	Letras, números y caracteres extraños.	No introducir el usuario o que éste no corresponda con el almacenado en la base de datos.
Contraseña	Letras, números y caracteres extraños.	No introducir la contraseña o que ésta no corresponda con la almacenada en la base de datos.

Caso de uso: Autenticarse
Caso de prueba: Realizar gestiones de autenticación entrando correctamente usuario y contraseña
Entrada: El usuario introduce correctamente el usuario y contraseña para autenticarse en el sistema. Usuario: danita Contraseña: *****
Resultado: Se le da acceso al usuario al sistema.
Condiciones: Los datos para autenticarse en el sistema deben estar en el rango de valores válidos.

Caso de uso: Autenticarse
Caso de prueba: Realizar gestiones de autenticación entrando incorrectamente usuario y contraseña o dejando los campos vacíos.
Entrada: El usuario introduce incorrectamente el usuario y la contraseña para autenticarse en el sistema o los deja vacíos. Usuario: " Acceso Denegado!! " Contraseña: *****
Resultado: El sistema muestra un mensaje de error.
Condiciones: Debe introducir los datos correctamente.

Tabla 4. 2 Caso de Prueba CU: Solicitar Cuenta.

Condición de Entrada	Casos Válidos	Casos no Válidos
Usuario	Letras, números y caracteres extraños.	No introducir el usuario o que el usuario introducido exista en la base de datos.
Contraseña	Letras, números y caracteres extraños.	No introducir la contraseña.
Confirmación de contraseña	Letras, números y caracteres extraños.	No introducir la confirmación de contraseña o que ésta no concuerde con la contraseña entrada anteriormente.
Nombre	Letras.	No introducir el nombre o introducir números o caracteres extraños.
Apellidos	Letras.	No introducir los apellidos o introducir números o caracteres extraños.
Correo	Letras, números y caracteres extraños, no puede faltar la @ y el punto.	No introducir el correo o introducir la dirección de correo sin la @ y el punto.

Caso de uso: Solicitar Cuenta.
Caso de uso: Solicitar la cuenta de usuario entrando correctamente todos los datos.
Entrada: El usuario introduce todos y de forma correcta los datos necesarios para solicitar una cuenta local en el sistema. Usuario: danita Contraseña: ***** Confirmación de contraseña: ***** Nombre: Danisleydi Apellidos: Noroña Blanco Correo: dnorona@estudiantes.uci.cu
Resultado: Se inserta la solicitud en la base de datos.
Condiciones: Los datos para solicitar una cuenta de usuario deben estar en el rango de valores válidos.

Caso de uso: Solicitar Cuenta.
Caso de prueba: Solicitar la cuenta de usuario entrando incorrectamente los datos o dejándolos vacíos.
Entrada: El usuario introduce incorrectamente alguno de los datos necesarios para solicitar una cuenta local en el sistema Usuario: danita Contraseña: ***** Confirmación de contraseña: ***** “ La contraseña escrita no corresponde” Nombre: Danisleydi Apellidos: Noroña Blanco 4 “Apellido incorrecto, solo puede contener letras” Correo: dnorona@estudiantes “ Dirección de correo no válida”
Resultado: El sistema muestra un mensaje de alerta notificando el error.
Condiciones: Debe introducir los datos correctamente.

Tabla 4. 3 Caso de Prueba CU: Editar Perfil.

Condición de	Casos Válidos	Casos no Válidos
Entrada		
Nombre	Letras.	No introducir el nombre o introducir números o caracteres extraños.
Apellidos	Letras.	No introducir los apellidos o introducir números o caracteres extraños.
Correo	Letras, números y caracteres extraños, no puede faltar la @ y el punto.	No introducir el correo o introducir la dirección de correo sin la @ y el punto.

Caso de uso: Editar Perfil.
Caso de uso: Editar el perfil de usuario entrando correctamente todos los datos.
Entrada: El usuario introduce todos y de forma correcta los datos necesarios para editar su perfil en el sistema. Nombre: Danisleydi Apellidos: Noroña Blanco Correo: dnorona@estudiantes.uci.cu
Resultado: Se actualizan los datos de usuario en la base de datos.
Condiciones: Los datos para solicitar una cuenta de usuario deben estar en el rango de valores válidos.

Caso de uso: Editar Perfil.
Caso de prueba: Editar el perfil de usuario entrando incorrectamente los datos o dejándolos vacíos.
Entrada: El usuario introduce incorrectamente alguno de los datos necesarios para editar su perfil en el sistema Nombre: Danisleydi Apellidos: Noroña Blanco 4 “Apellido incorrecto, solo puede contener letras” Correo: dnorona@estudiantes “ Dirección de correo no válida”
Resultado: El sistema muestra un mensaje de alerta notificando el error.
Condiciones: Debe introducir los datos correctamente.

4.4 Conclusiones

En este capítulo se presentaron los componentes necesarios para llevar a cabo la implementación del sistema, así como el diagrama de despliegue. Además se presentaron los casos de prueba de caja negra realizadas al sistema.

CONCLUSIONES.

Con la culminación del presente trabajo de diploma, se obtuvieron los resultados que se detallan a continuación.

Se realizó un estudio profundo de las principales formas de administración de sistemas así como las herramientas reporteadoras más utilizadas en la actualidad, obteniéndose así la información necesaria para poder llevar a cabo los módulos a desarrollar.

Se seleccionaron las herramientas necesarias para realizar el proyecto con la calidad requerida y que estuvieran a tono con las disposiciones existentes, siempre que fuera posible, sobre el uso de software libre y multiplataforma.

Se obtuvo el módulo Administración que posibilita que el Sistema de Análisis Estadístico de Datos sea mucho más rápido en el procesamiento de datos, así como una mayor organización de los permisos de usuarios.

Se obtuvo además el módulo Reporte, mediante el cual se les brinda a los usuarios la posibilidad de obtener información de las encuestas que tienen publicadas en el momento que la necesiten.

Por tanto, podemos concluir que se logró satisfactoriamente el cumplimiento del objetivo propuesto para el presente trabajo de diploma.

RECOMENDACIONES.

A continuación se muestran un conjunto de recomendaciones con el objetivo de que el trabajo propuesto sea utilizado por el mayor número de personas a las que pueda ser de utilidad.

1. La facultad 8 se encargue de gestionar la ampliación del módulo de análisis estadístico a todas las técnicas y pruebas que sean aplicables a una encuesta.
2. El presente trabajo sea utilizado no sólo en la facultad 8 sino por todo aquel que requiera de su utilización en la UCI.
3. La facultad 8 negocie, con empresas u organismos nacionales, la utilización del sistema y de ser posible su comercialización.

Referencias Bibliográficas.

(2001). "Lenguajes de Programación: "Programación Web"." from <http://lenguajes-de-programacion.com/programacion-web.shtml>

(2005, 17/02). "Manual de referencia de MySQL." from www.mysql.com. .

(2007). "Lighttpd." Retrieved 1 de febrero de 2008, from <http://libertonia.escomposlinux.org/story/2007/4/18/12850/2410>.

Barroso, A. (2006). Sistema Automatizado del Registro Mercantil de Cuba: SAREMC. Cuba.

Booch, G. R., J. y Jacobson (2000). ""El Lenguaje Unificado de Modelado"." I: 11.

Directo, D. d. M. (1999-2008). "Personalización." Retrieved 16/03/08, from http://www.marketingdirecto.com/glosario/datos_termino.php?termino=Personalizaci%F3n.

Felipe, D. R. (02-2004). "Manual de Desarrollo Web: ¿Qué es cada tecnología?" from <http://www.desarrolloweb.com/manuales/15/>.

Jim, C. (Octubre 04, 2002). "Building Web Applications with UML".

Kabir, M. J. (2003). "La biblia de Servidor Apache 2." **Vol: I**.

Lascano, E. (2007). "Desarrollo de aplicaciones de cliente servidor en dos, tres y n capas, metodologías y tecnologías." from http://www.estrategiasysoluciones.com/archivos/aplicaciones2_3_n_capas.pdf.

Manso, R. L., Marylyn (2000). ""Aplicación del Formato Dublin Core para la descripción de los recursos en el Sistema de Gestión Multimedia desarrollado en el CDICT- UCLV"."

Martin Enrique, S. F. (2002). Sistema ANACONDA para el análisis automático de la calidad del software.

Observatorio, R. e. (2008). "Glosarios de red.es observatorio." Retrieved 15/03/08, from http://iteso.mx/~lilianar/materiales/reporte_de_investigacion.htm.

PROAÑO, I. D. J. B. (2006). "ANALISIS COMPARATIVO DE BASES DE DATOS DE CODIGO ABIERTO VS CODIGO CERRADO (DETERMINACION DE INDICES DE COMPARACION)." Retrieved 1 de febrero del 2008, from <http://www.mysql-hispano.org/articulos/num43/analisis-comparativo.pdf>.

Productos, E. R. H. d. D. d. "Red Hat Enterprise Linux 4: Introducción a la administración de sistemas." Retrieved 10/02/08, from <http://web.mit.edu/rhel-doc/4/RH-DOCS/rhel-isa-es-4/ch-philosophy.html>.

Ruvalcaba, L. G. "Reporte de investigación." Retrieved 12 de diciembre de 2007, from http://iteso.mx/~lilianar/materiales/reporte_de_investigacion.htm.

Saether Bakken, S. S., Egon. (2003). "Manual de Php. PHP Documentation Group." from <http://www.php.net/docs.php>.

Sánchez, M. A. M. (2004). Metodologías De Desarrollo De Software.

Bibliografía

2006. Ciberaula. *Ciberaula*. [Online] 2006. [Cited: enero 10, 2008.] http://linux.ciberaula.com/articulo/linux_apache_intro/.

2007. *Ejemplo básico con Jasper Report*. [Online] agosto 6, 2007. [Cited: febrero 10, 2008.] http://www.chuidiang.com/chuwiki/index.php?title=Ejemplo_b%C3%A1sico_con_Jasper_Report.

2008. Administrar los usuarios de sitios Web con funciones. *Administrar los usuarios de sitios Web con funciones*. [Online] 2008. [Cited: febrero 2, 2008.] [http://msdn2.microsoft.com/es-es/library/t32yf0a9\(VS.80\).aspx](http://msdn2.microsoft.com/es-es/library/t32yf0a9(VS.80).aspx).

Jacobson, Ivar, Booch, Grady and Rumbaugh, James. 2000. *El proceso unificado de desarrollo del software*. Madrid : Pearson Educación, S.A, 2000.

Joomla! para principiantes. *Joomla! para principiantes*. [Online] [Cited: febrero 4, 2008.] http://e-book.joomlaos.net/cap_06.pdf.

Larman, Craig. 1999. *UML y Patrones*. México : Prentice Hall, 1999.

Manager, Equipo de Free Download. Free Download Manager. *Rpv Software*. [Online] [Cited: febrero 6, 2008.] http://www.freedownloadmanager.org/es/downloads/Informes_de_Rpv_51292_p/.

Rumbaugh, James, Jacobson, Ivar and Booch, Grady. *El lenguaje unificado de modelado. Manual de referencia*.

Sánchez, Aliuska and Arias, Bettys L. 2007. *Sistema de Análisis Estadístico de Datos*. Ciudad Habana : s.n., 2007.

Schmuller, Joseph. 2000. *Aprendiendo UML en 24 horas*. México : Pearson Educación, 2000.

Software, Equipo Tivoli. Tivoli Software. *Administración delegada de roles*. [Online] [Cited: febrero 2, 2008.] http://publib.boulder.ibm.com/tividd/td/ITAME/SC32-1360-00/es_ES/HTML/am51_admin268.htm.

Glosario de términos.

Administrador de sistemas: Persona encargada de un sistema informático, tanto de su instalación como de su puesta a punto y mantenimiento, debiendo velar porque todo funcione de acuerdo con lo esperado. Además, el administrador tiene que hacer frente a cuantas incidencias se produzcan en el sistema, y establecer mecanismos de seguridad.

Apache: Servidor de páginas web de código abierto para diferentes plataformas (UNIX, Windows, etc.)

API: *Application Programming Interface*, Interfaz de Programación de Aplicaciones. Conjunto de especificaciones de comunicación entre componentes software; conjunto de llamadas al sistema que ofrecen acceso a los servicios del sistema desde los procesos y representa un método para conseguir abstracción en la programación, generalmente (aunque no necesariamente) entre los niveles o capas inferiores y los superiores del software.

Aplicación: Programa con el que el usuario final interactúa a través de una interfaz, realizando tareas útiles para éste.

Arquitectura software: Conjunto de patrones y abstracciones coherentes que proporcionan el marco de referencia necesario para guiar la construcción del software para un sistema de información. Establece los fundamentos para que los desarrolladores trabajen en una línea común que permita alcanzar los objetivos y necesidades del sistema.

Capa de Datos: Es donde residen los datos. Está formada por uno o más gestores de bases de datos que realizan todo el almacenamiento de información, reciben solicitudes de almacenamiento o recuperación de información desde la capa de negocio.

Capa de Negocio: Es donde residen los programas que se ejecutan, recibiendo las peticiones del usuario y enviando las respuestas tras el proceso. Se establecen todas las reglas que deben cumplirse. Esta capa se comunica con la capa de presentación, para recibir las solicitudes y presentar los resultados, y con la capa de datos, para almacenar o recuperar datos.

Capa de Presentación: Presenta el sistema al usuario, le comunica información y captura información de éste dando un mínimo de proceso (realiza un filtrado previo para comprobar que no hay errores de formato). Esta capa se comunica únicamente con la capa de negocio.

Componente: Parte física y reemplazable del sistema que cumple y proporciona la realización de un conjunto de interfaces, ejemplo: ficheros de código fuente, scripts, ficheros de código binario, ejecutables y similares.

Despliegue: Cuando varios trabajos relativamente independientes (flujos de control, procesos) se distribuyen entre diferentes dispositivos hardware (procesadores).

Encuesta: Es el documento que contiene las preguntas formuladas por el encuestador y que el encuestado debe contestar

Encuestador: Persona encargada de elaborar la encuesta, aplicarla y solicitar y obtener los reportes.

Estadígrafo: Es la medida que en estadística se aplica sobre una muestra.

Estadística: La Estadística es la rama de las matemáticas que describe, analiza e interpreta ciertas características de un conjunto de individuos llamado población.

Herramienta: Software que se utiliza para automatizar las actividades definidas en el proceso.

Herramienta CASE: *Computer Aided Software Engineering*, Ingeniería de Software Asistida por Ordenador. Diversas aplicaciones informáticas destinadas a aumentar la productividad en el desarrollo de software reduciendo el costo de las mismas en términos de tiempo y de dinero.

HTML: Lenguaje basado en marcas que indican las características del texto, utilizado para definir documentos de hipertexto en webs.

HTTP: Protocolo cliente-servidor utilizado para el intercambio de páginas web.

Investigador: Persona que ejecuta acciones sistemáticas orientadas a la creación y generación de nuevo conocimiento.

Ingeniería de Software: Aplicación de un enfoque sistemático, disciplinado y cuantificable al desarrollo, operación y mantenimiento del software.

MySQL: Base de datos relacional multiplataforma de código abierto, muy utilizada en aplicaciones web.

Open Source: 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 software libre.

PHP: Lenguaje de programación para el desarrollo de webs dinámicas, originalmente se conocía como Personal Home Page tools, herramientas para páginas personales.

Plataforma de Desarrollo: Entorno común en el cual se desenvuelve la programación de un grupo definido de aplicaciones. Comúnmente se encuentra relacionada directamente a un sistema operativo, sin embargo, también es posible encontrarla ligada a una familia de lenguajes de programación.

Proceso de Desarrollo: Definición del conjunto completo de actividades necesarias para transformar los requisitos de un usuario en un producto.

Producto: Artefactos que se crean durante la vida del proyecto, como los modelos, código fuente, ejecutables y documentación.

Reporte: Informe que se emite o se presenta por una persona, entidad o sistema.

Reporteadores: Herramientas especializadas en la construcción de reportes que son capaces de explotar la información de las bases de datos para construir dichos reportes.

Requisito: Condición o capacidad que debe cumplir un sistema.

Requisito Funcional: Especifica una acción que debe ser capaz de realizar el sistema sin considerar las restricciones físicas. Especifica el comportamiento de entrada/salida de un sistema.

RUP: Es un proceso en el que se han unificado técnicas de desarrollo y metodologías, está preparado para desarrollar grandes y complejos proyectos y es orientado a objetos.

Seguridad: Asegurar que los recursos del sistema de una organización sean utilizados de la manera que se decidió y que la información que se considera importante no sea fácil de acceder por cualquier persona que no se encuentre acreditada.

SGBD: Un Sistema de Gestión de Base de Datos (SGBD) es el software que permite la utilización y actualización de los datos almacenados en una (o varias) base(s) de datos.

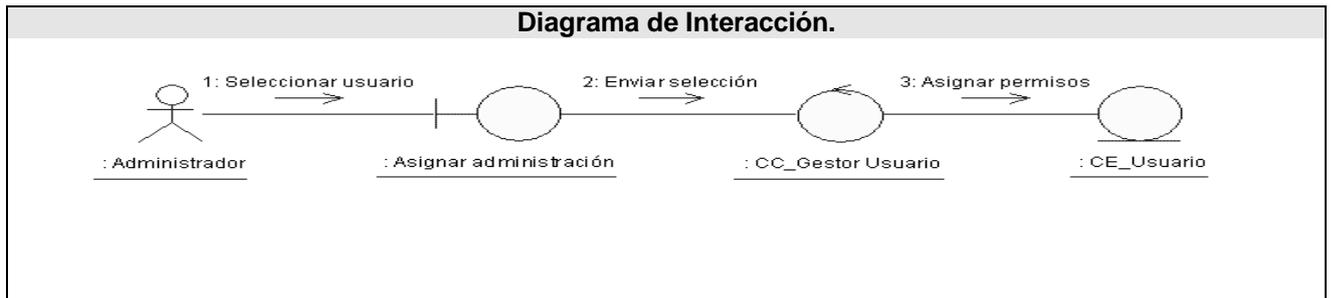
URL: Localizador uniforme de recurso (Uniform Resource Locator). Es una secuencia de caracteres, de acuerdo a un formato estándar, que se usa para nombrar recursos, por su localización.

Web: Sistema para presentar información en Internet basado en hipertexto. Cuando se utiliza en masculino (*el web, un web*) se refiere a un sitio web entero, en cambio si se utiliza en femenino (*la web, una web*) se refiere a una página web concreta dentro del sitio web.

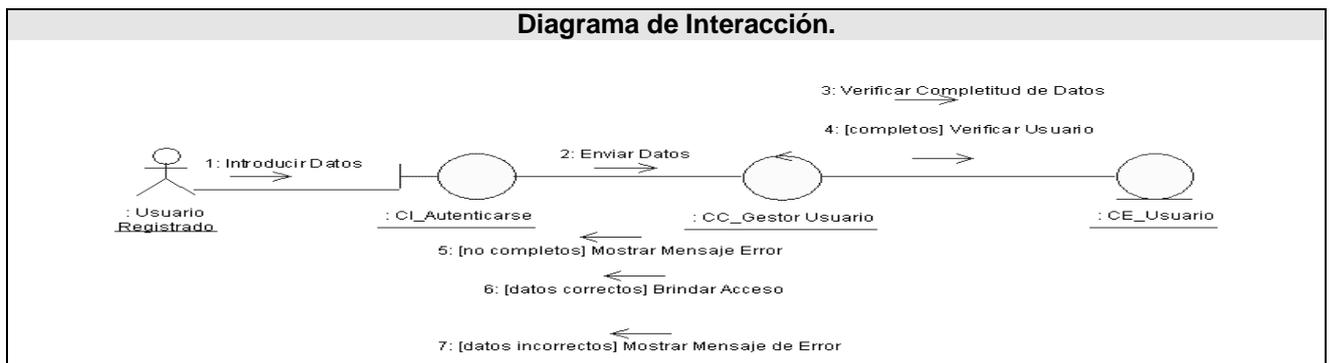
ANEXOS

Anexo # 1 Diagramas de colaboración del análisis.

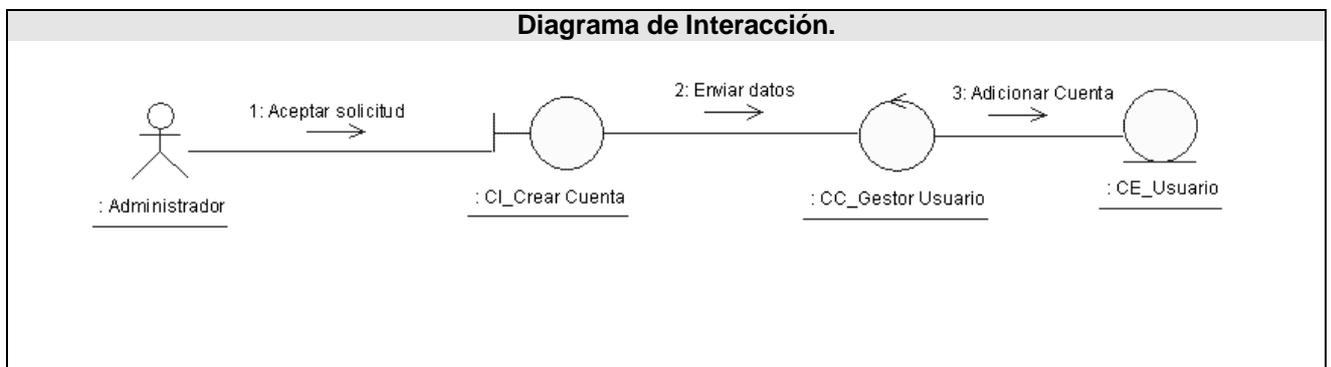
1.1 Caso de Uso Asignar Administración



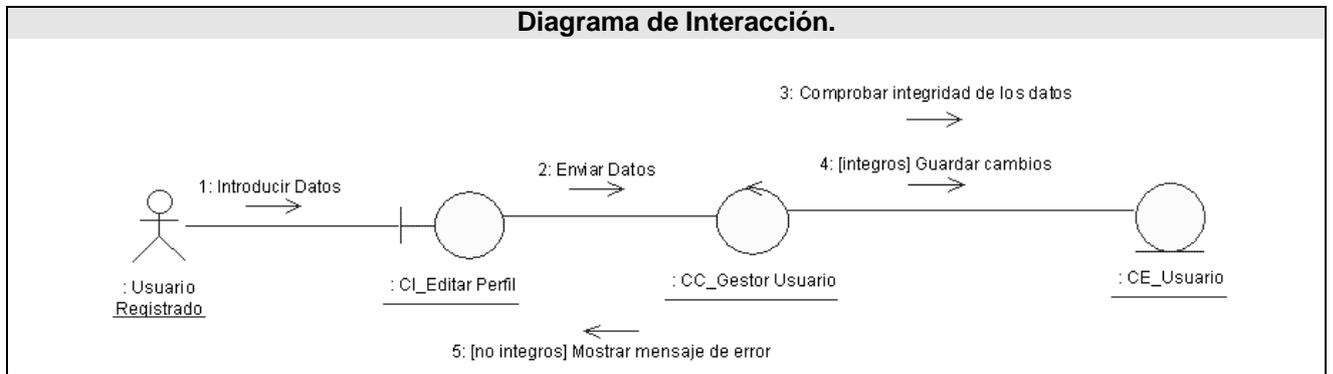
1.2 Caso de Uso Autenticar Usuario



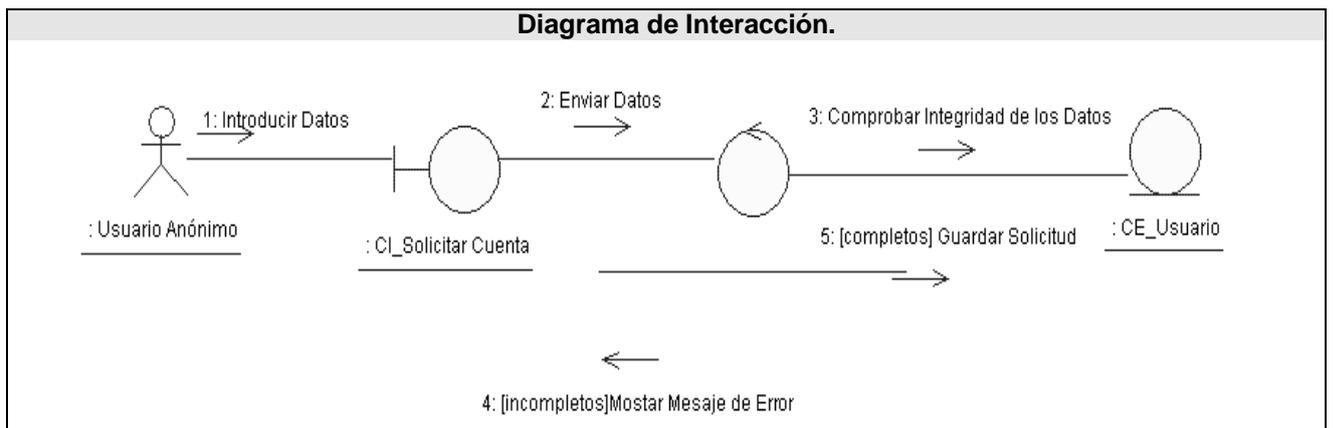
1.3 Caso de Uso Crear Cuenta



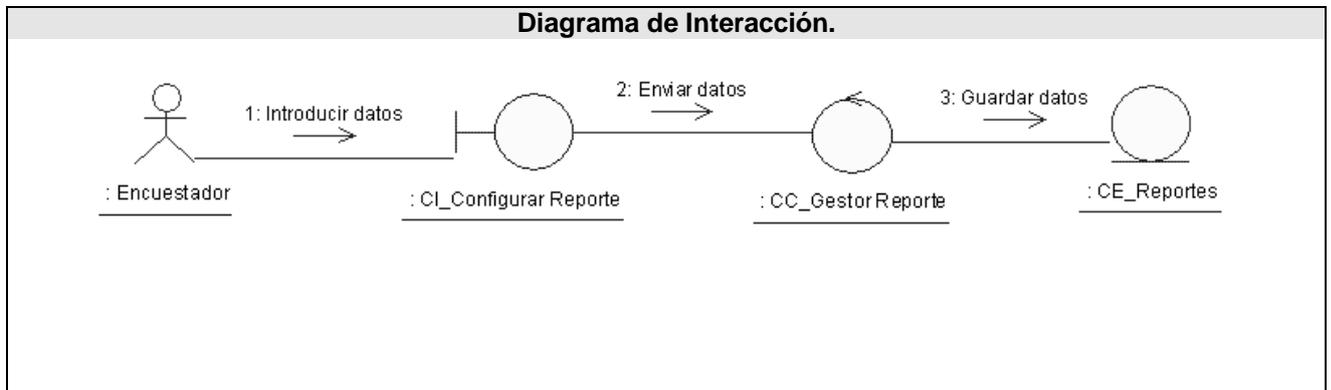
1.4 Caso de Uso Editar Perfil



1.5 Caso de Uso Solicitar Cuenta

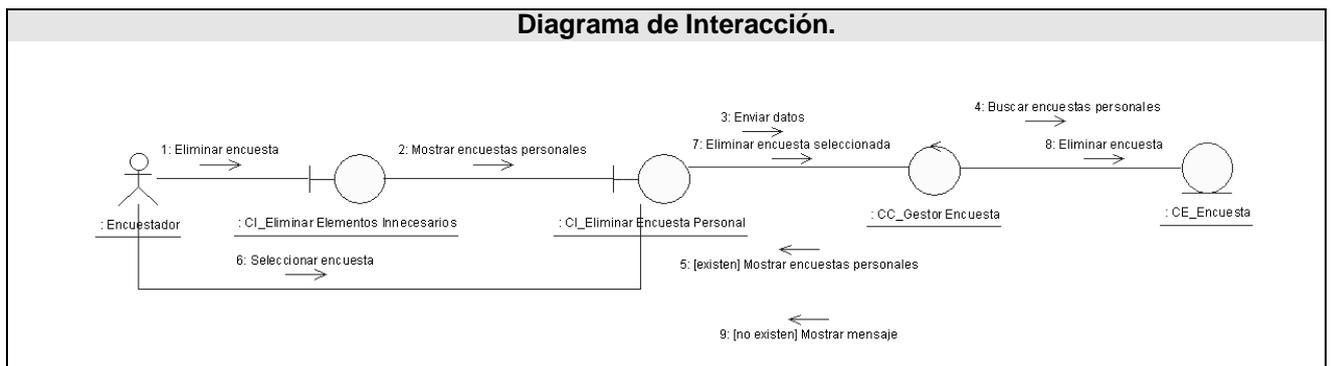


1.6 Caso de Uso Configurar Reporte

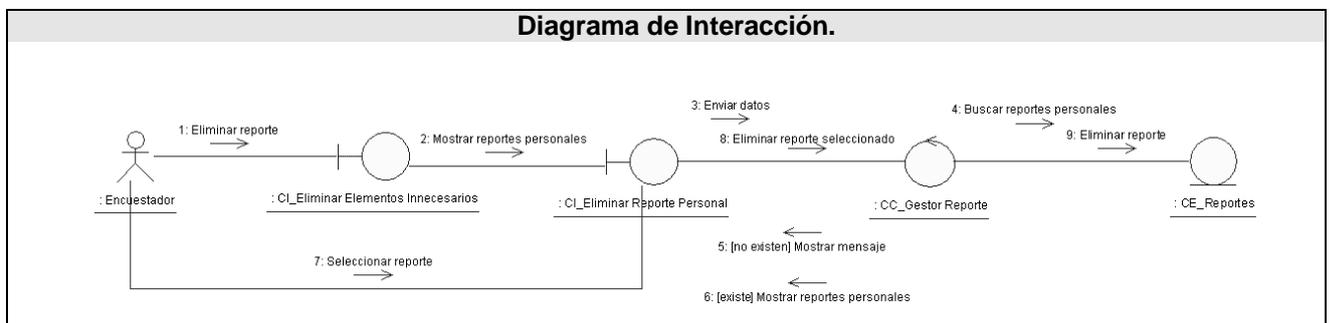


1.7 Caso de Uso Eliminar Elementos Innecesarios

1.7.1 Escenario Eliminar Encuestas

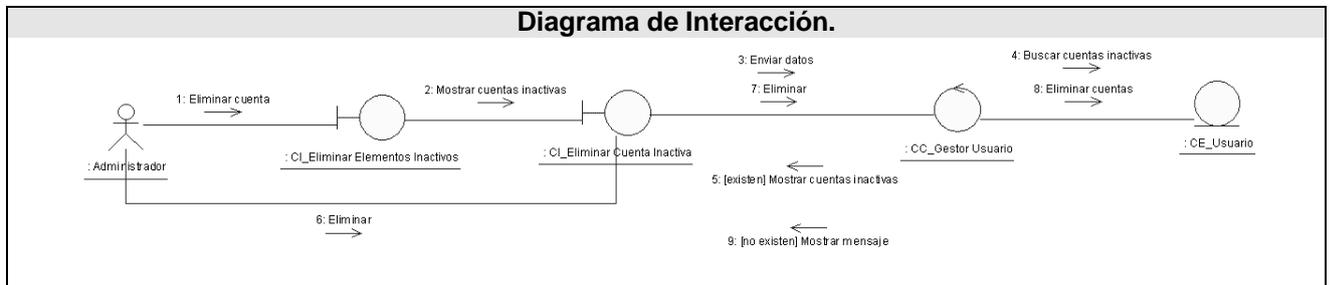


1.7.2 Escenario Eliminar Reportes

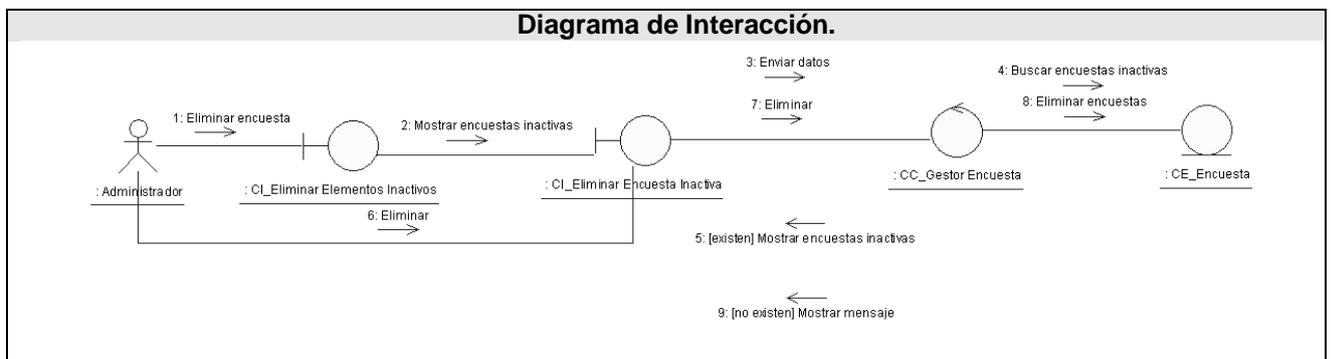


1.8 Caso de Uso Eliminar Elementos Inactivos

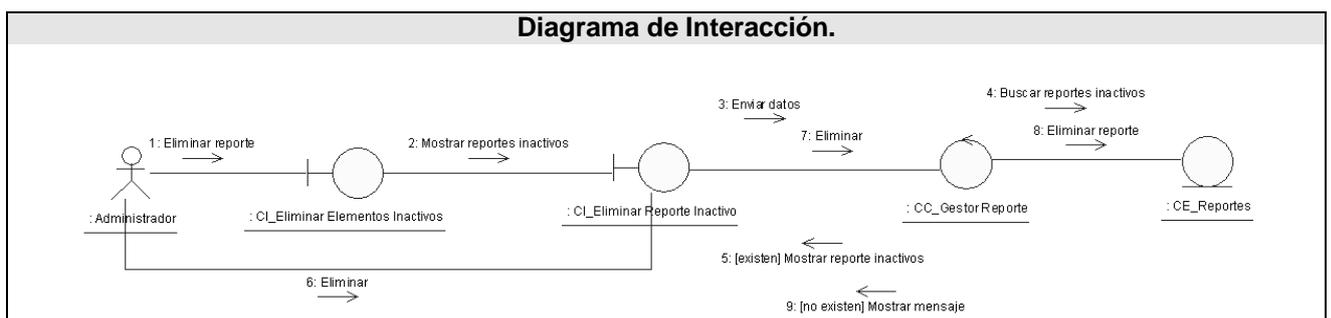
1.8.1 Escenario Eliminar Cuentas



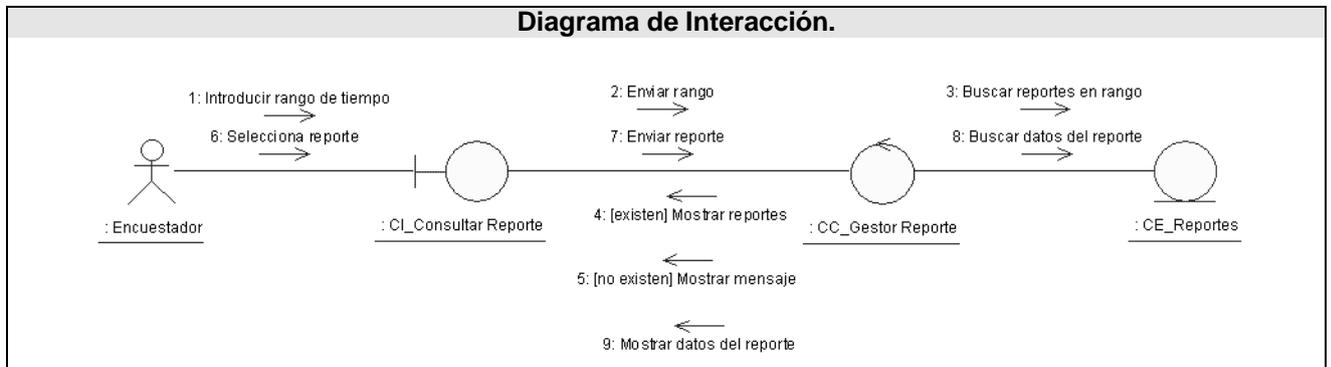
1.8.2 Escenario Eliminar Encuestas



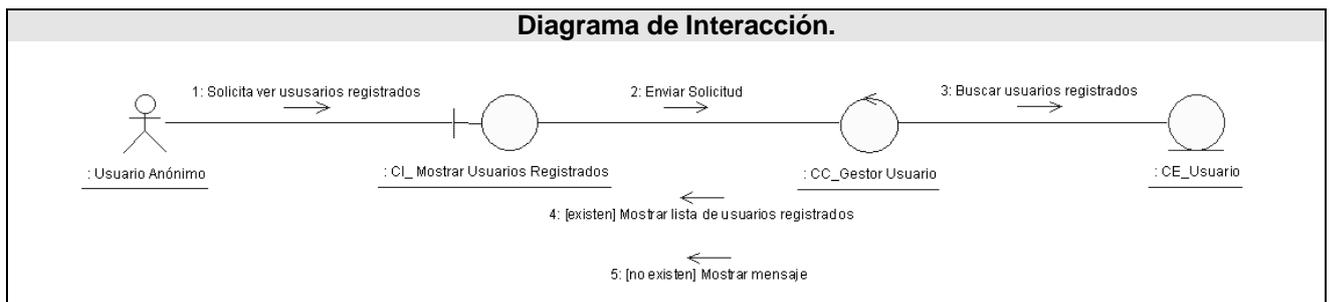
1.8.3 Escenario Eliminar Reportes



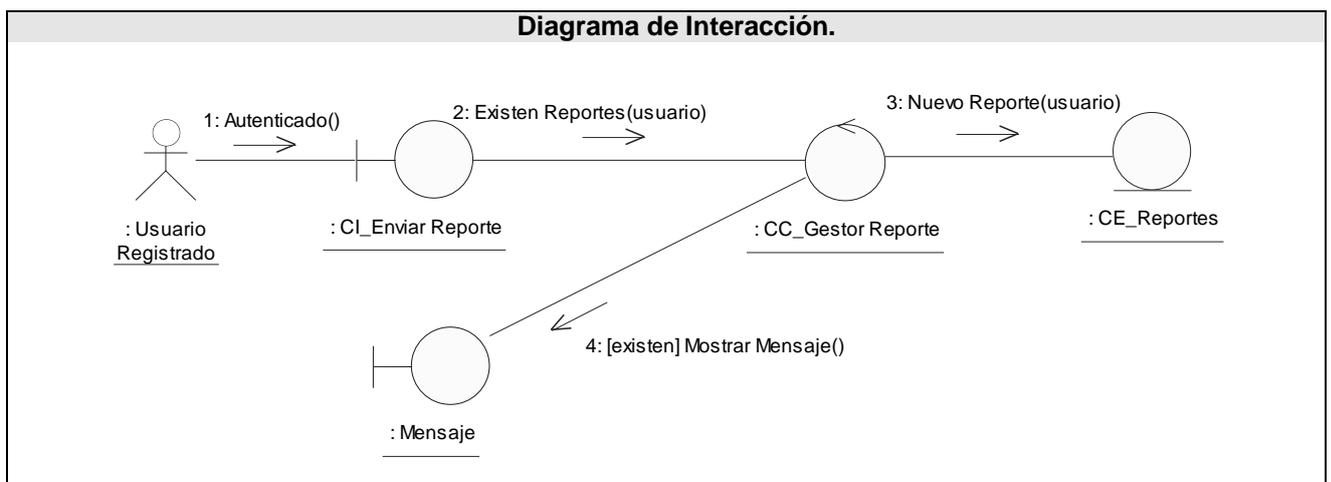
1.9 Caso de Uso Consultar Reporte



1.10 Caso de Uso Mostrar Usuarios Registrados

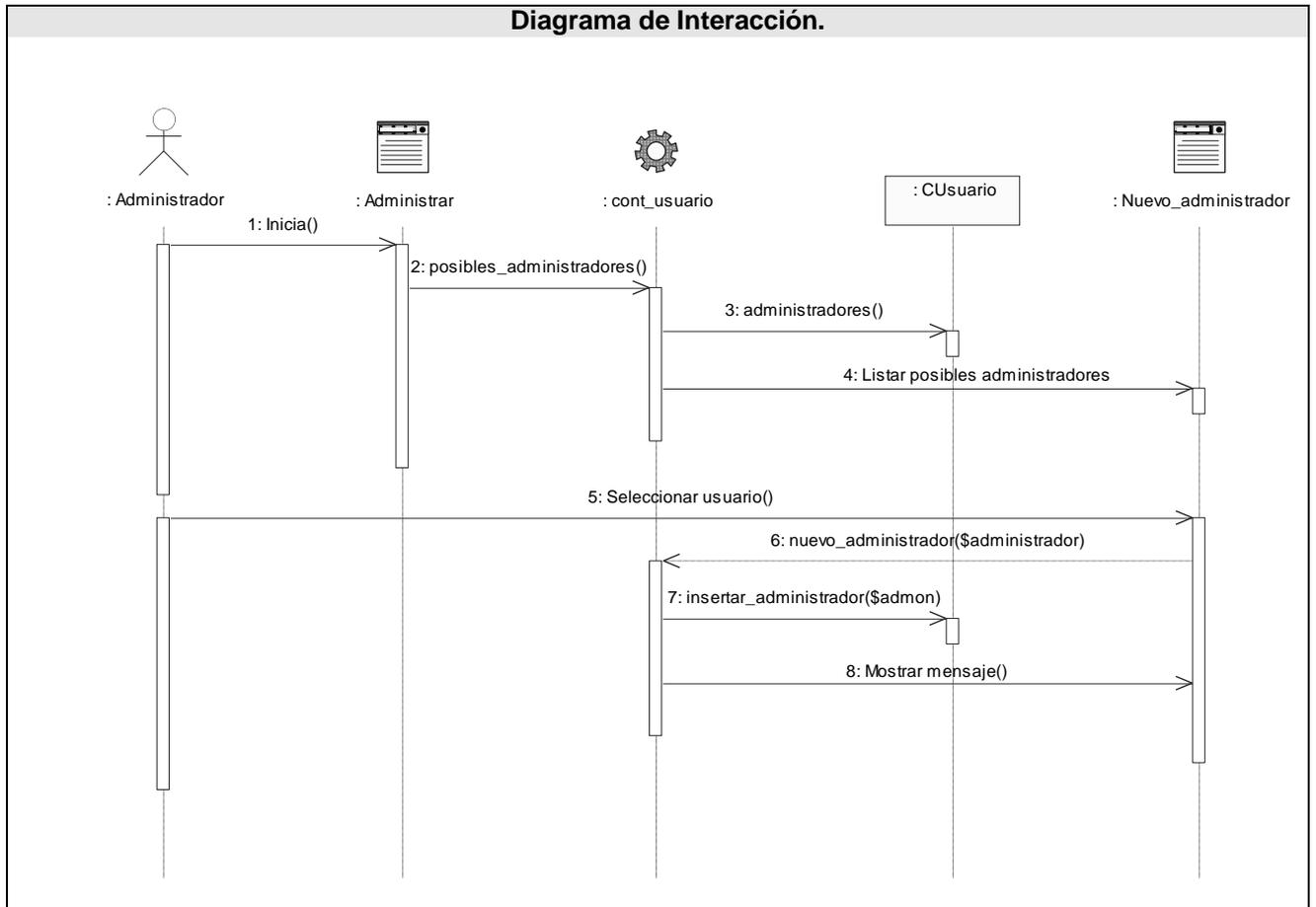


1.11 Caso de Uso Enviar Reporte

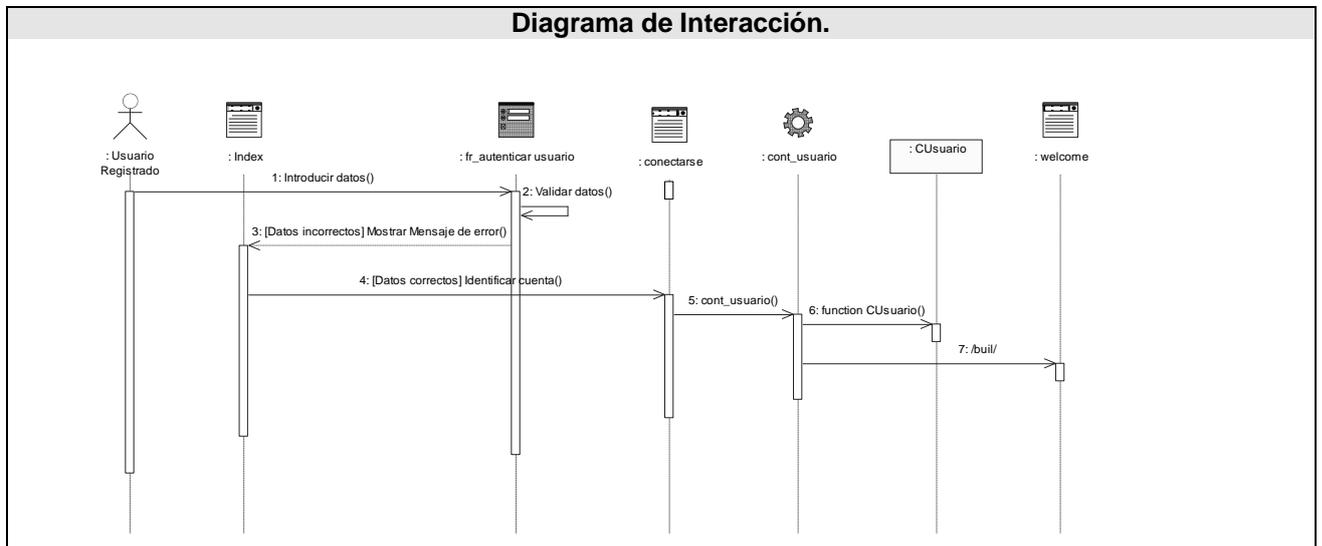


Anexo # 2 Diagramas de secuencias del diseño

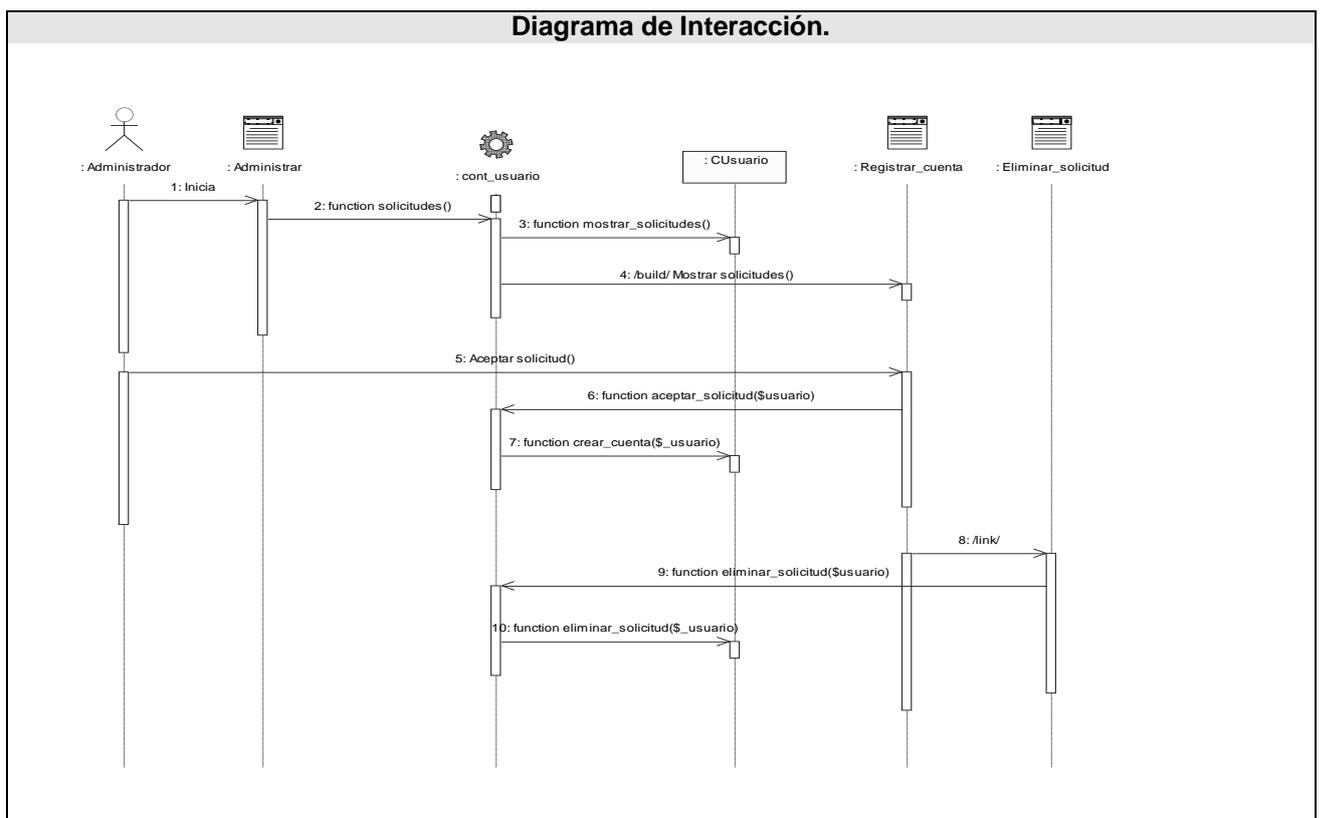
2.1 Caso de Uso Asignar Administración



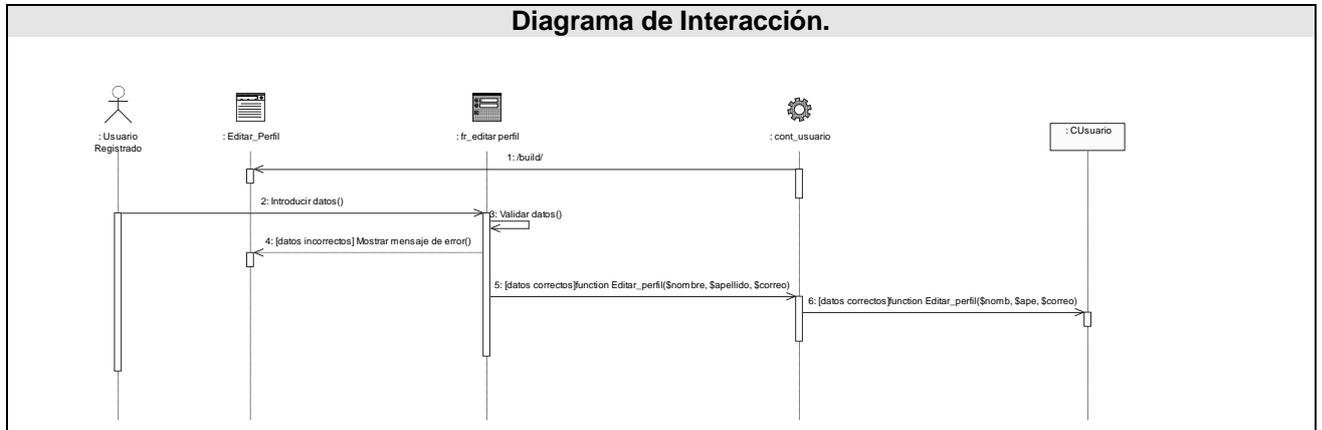
2.2 Caso de Uso Autenticar Usuario



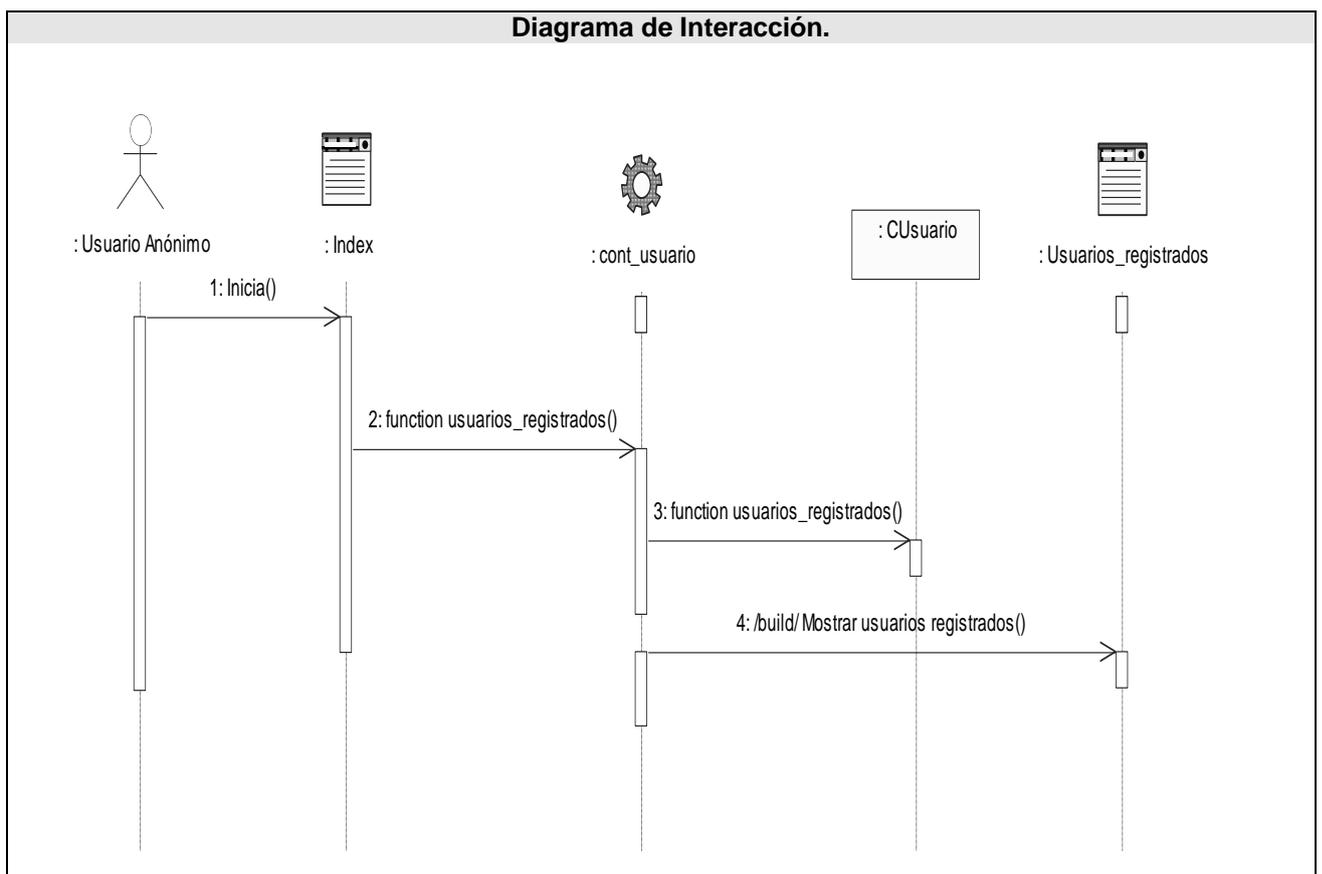
2.3 Caso de Uso Crear Cuenta



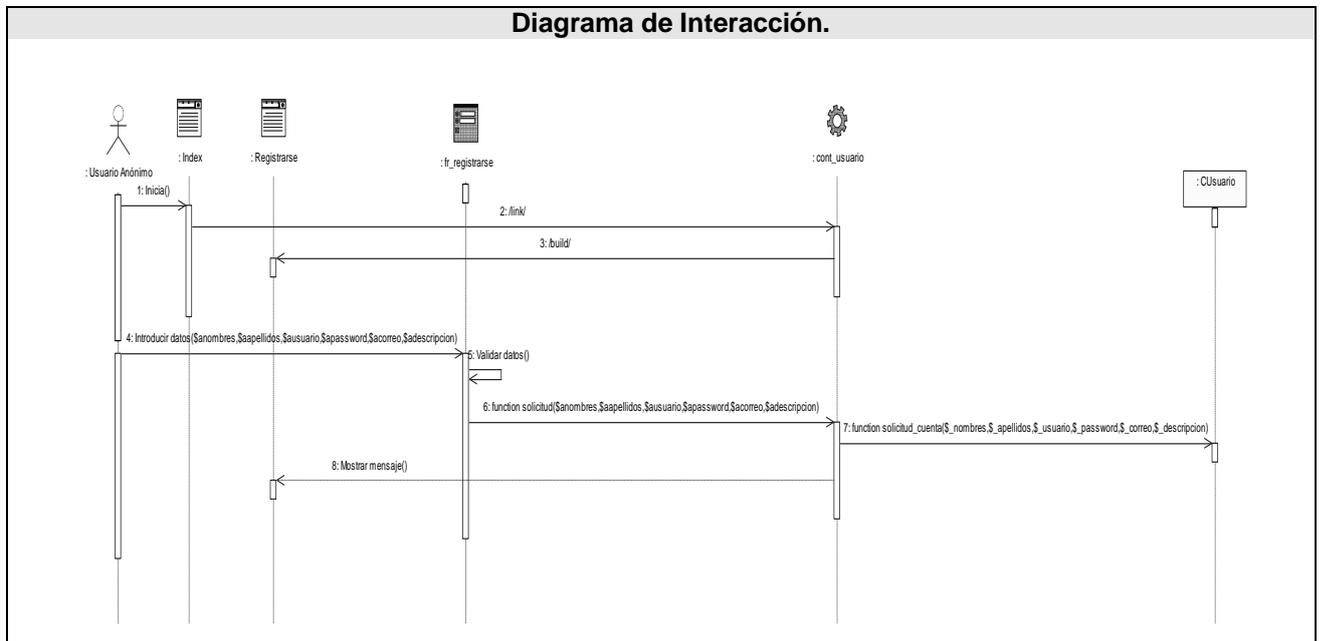
2.4 Caso de Uso Editar Perfil



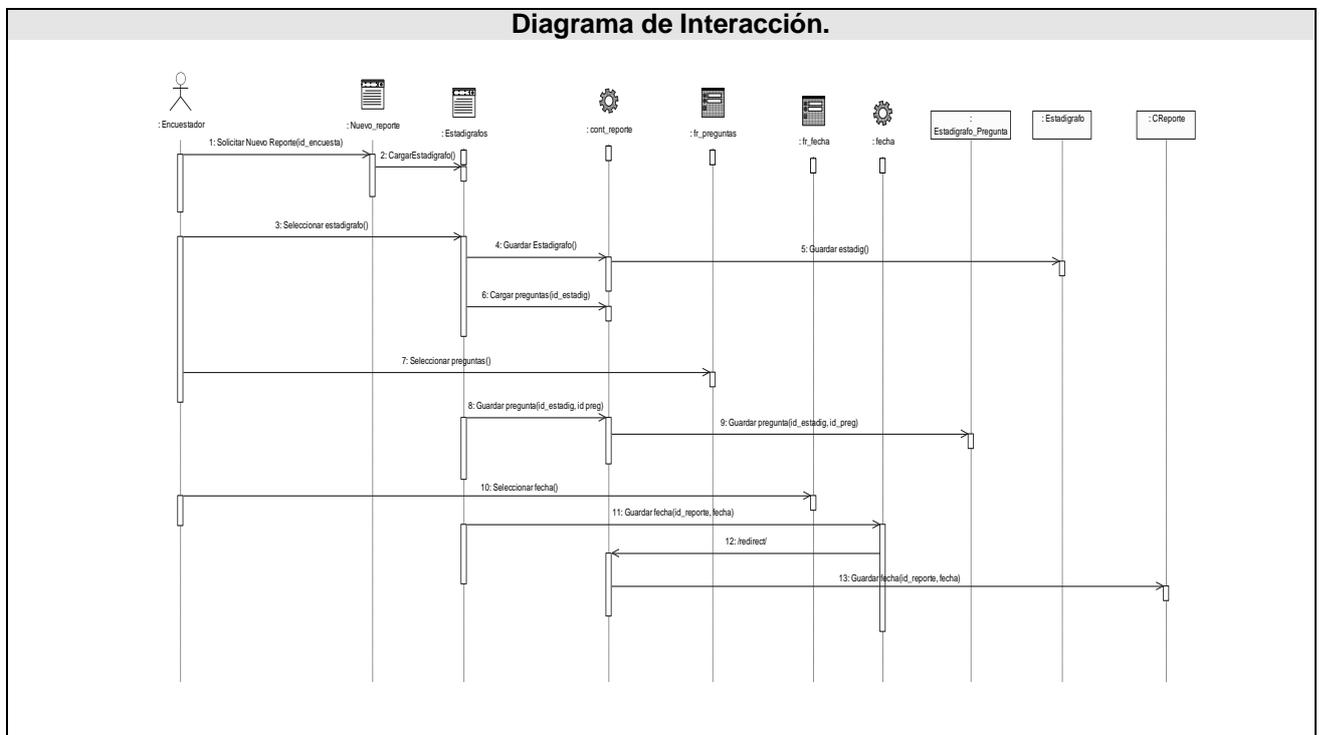
2.5 Caso de Uso Mostrar Usuarios Registrados



2.6 Caso de Uso Solicitar Cuenta

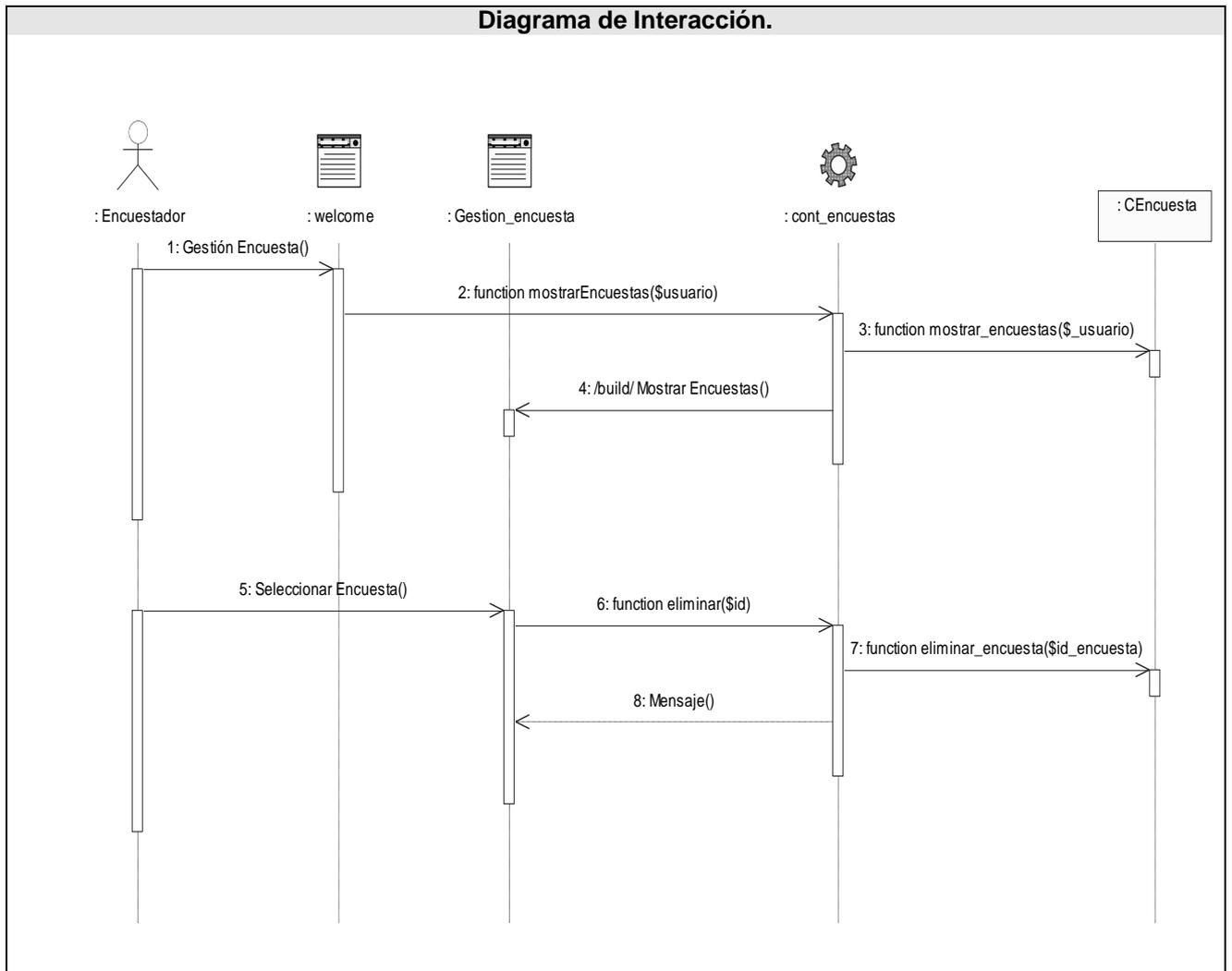


2.7 Caso de Uso Configurar Reporte



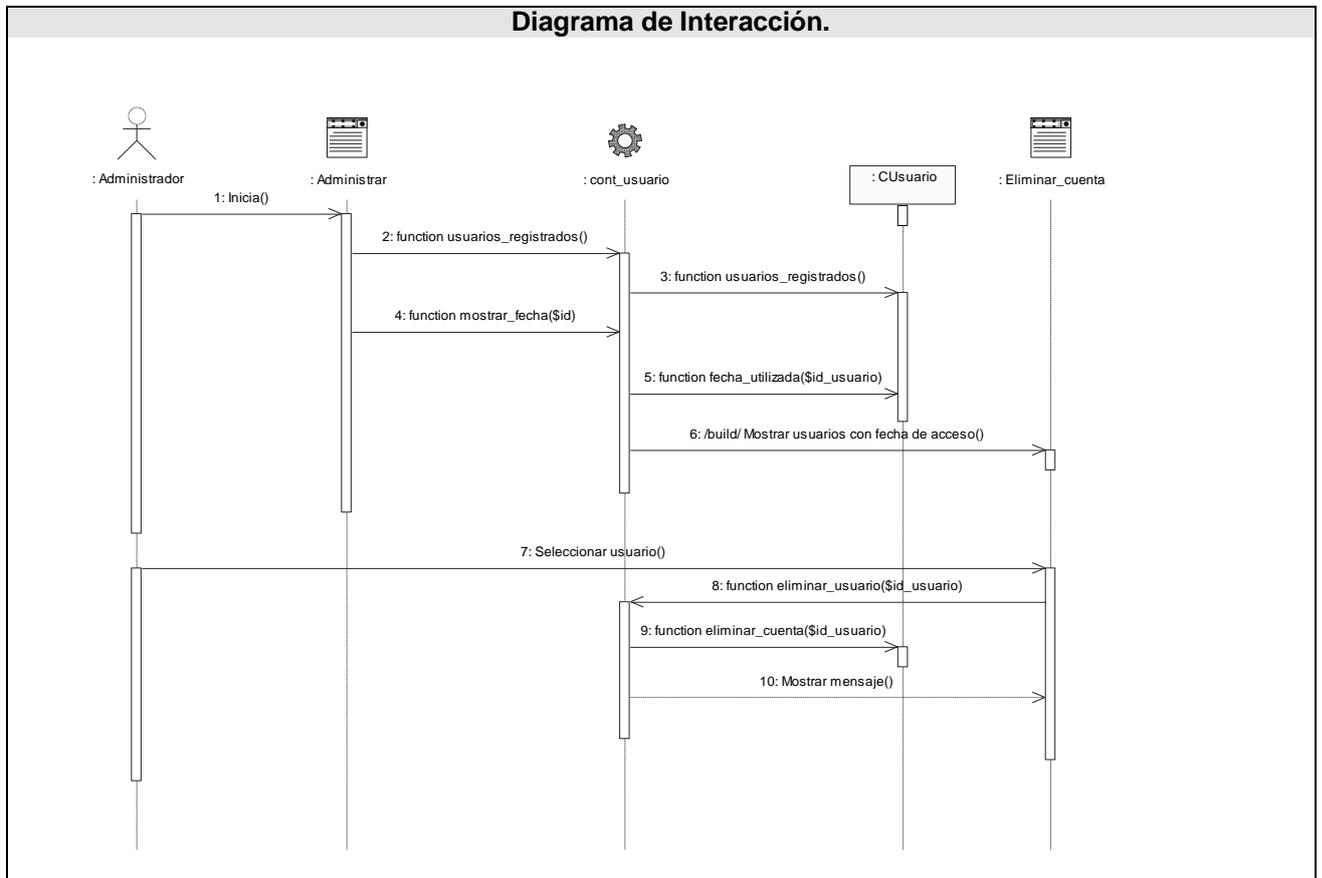
2.8 Caso de Uso Eliminar Elementos Innecesarios

2.8.1 Escenario Eliminar Encuestas



2.9 Caso de Uso Eliminar Elementos Inactivos

2.9.1 Escenario Eliminar Cuentas



2.9.2 Escenario Eliminar Encuestas

