

*Universidad de las Ciencias Informáticas
Facultad 9*



***Título: CEVALE – CEntro Virtual de Auto
aprendizaje de Lenguas Extranjeras.
Módulo de Administración***

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

Autores: *Karelia Cabada Jiménez*

Yuliet Leyva López

Tutor: *Ing. Yurisbel Vega Ortiz*

Co-Tutor: *M.Sc. Yareisis Pérez Heredia*

Ciudad de La Habana, Cuba

Julio de 2008

Año 50 de la Revolución

DEDICATORIA

A mi madre por confiar en mí siempre, por amarme, educarme y apoyarme en todo lo correcto, ensañándome a esforzarme y no amilanarme ante las dificultades. Quien ha dado con amor todo cuanto ha podido para que lograra este éxito que hoy es de nosotras.

A mi tía Angela que es una persona muy especial en mi vida, que me ha dado siempre su amor y nunca ha dejado de preocuparse por mí a pesar de estar muy lejos y aún así, en los últimos momentos sus consejos han sido fundamentales contra el cansancio, el desánimo.

A mi hermana Kirenia por ser la hermana más maravillosa que la vida me pudiera haber dado y por formar parte de lo que más quiero en el mundo.

A mis tres sobrinos por su amor y cariño...

A mi abuela Blanca que siempre ha estado presente con sus sabias palabras...

A Lester por ser como un padre, por ser incondicional, por ser único en mi vida...

A mi familia que se ha preocupado por mí en todo momento y han ayudado a conquistar este logro.

Karelia.

A mis adorados padres por ser mis guías, mis confidentes, mi razón de ser, lo más grande que tengo en esta vida y a mi hermano por ser tan lindo como él es.

A mi novio por su comprensión y apoyo incondicional.

Yuliet.

AGRADECIMIENTOS

Quisiéramos agradecerles a todos los que, incondicionalmente, depositaron todo su empeño y confianza en nosotras y nos tendieron su mano cuando el camino se tornaba más áspero, de forma especial a:

A nuestros padres por su amor e incondicionalidad.

A nuestras familias.

A nuestro tutor, Yuribel Vega Ortiz, por su incondicionalidad, su paciencia y su ayuda desinteresada.

A Yareisis, nuestra jefa de proyecto que dedicó gran parte de su tiempo ayudándonos durante el desarrollo de este trabajo.

A nuestros profesores por inculcarnos tan valiosos conocimientos en estos cinco años.

A nuestros compañeros que durante cinco importantes años de nuestras vidas compartimos momentos inolvidables.

A Fidel por haber tenido la maravillosa idea de crear esta universidad.

También a todas aquellas personas que en momentos de desaliento nos brindaron su apoyo, su ayuda y asesoría.

A todos muchas gracias.

DECLARACIÓN DE AUTORÍA

Declaramos que Karelia Cabada Jiménez y Yuliet Leyva López son las únicas autoras de este trabajo y autorizan a la Universidad de las Ciencias Informáticas a hacer uso del mismo en su beneficio.

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

Karelia Cabada Jiménez.

Yuliet Leyva López.

Ing. Yurisbel Vega Ortiz.

OPINIÓN DEL TUTOR

El tutor del presente Trabajo de Diploma considera que durante su ejecución los estudiantes mostraron las cualidades que a continuación se detallan:

Ambos mostraron una muy alta responsabilidad, laboriosidad, independencia y un gran sentido del compromiso laboral, durante todo el desarrollo de su Trabajo de Diploma; factores que permitieron alcanzar con éxito los objetivos trazados. Los diplomantes dieron fe de consagración y disciplina, además de un alto grado de interés y motivación por desarrollar procesos investigativos serios, novedosos y creativos.

El documento cumple con los requisitos de calidad científico-técnicos establecidos para dichos trabajos y los resultados obtenidos presentan elevado valor para los CASIE de la Universidad de las Ciencias Informáticas, aún cuando el alcance de esta investigación solo llegará hasta el diseño de la solución propuesta.

Por todo lo anteriormente expresado se considera que los estudiantes están aptos para ejercer como Ingenieros en Ciencias Informáticas; y se propone al tribunal que se le otorgue al Trabajo de Diploma la calificación de 5 puntos.

Se firma la presente a los _____ días del mes de _____ del 2008.

Ing. Yurisbel Vega Ortiz.

RESUMEN

La Universidad de las Ciencias Informáticas se ha impuesto la tarea de crear un proyecto enmarcado en la obtención de un sistema capaz de gestionar los servicios que se brindan en los CASIE, desde el asesoramiento de los usuarios por parte de los profesores de los distintos idiomas extranjeros que en la Universidad se imparten hasta el préstamo de materiales que sirven de apoyo al aprendizaje de un idioma determinado. Para la realización del sistema CEVALE se determinó dividir en 6 módulos el proyecto.

La presente investigación se encarga de desarrollar el Módulo de Administración, quien se responsabiliza de la autenticación de los usuarios, la asignación de roles y la seguridad de la información, con el fin de lograr mantener la integridad, confiabilidad y disponibilidad de la misma, teniendo como objetivo general el diseño de dicho módulo.

Para la modelación de los procesos del sistema se utiliza como metodología, el Proceso Unificado de Desarrollo (RUP) y el lenguaje de modelado UML, y para la construcción se utiliza como lenguaje de programación: PHP5, IDE de desarrollo: Eclipse, y gestor de base de datos: MySQL.

Con la propuesta se pretende convertir al sistema en general en un producto sólido, fiable y extensible, logrando la integridad y consistencia de la información procesada.

PALABRAS CLAVES

- CEVALE
- Aplicación Web
- Autenticación
- Control de Usuario

TABLAS Y FIGURAS

Tabla 1. Actores del negocio	34
Tabla 2. Trabajadores del negocio.....	35
Tabla 3. Descripción textual del CUN_ Atender Usuario.	38
Tabla 4. Descripción textual del CUN_ Realizar Encuesta	38
Tabla 5. Descripción textual del CUN_ Solicitar Asesoría	39
Tabla 6. Descripción textual del CUN_ Seguir ruta de aprendizaje.....	40
Tabla 7. Descripción textual del CUN_ Solicitar recurso.....	40
Tabla 8. Descripción textual del CUN_ Localizar recurso.....	42
Tabla 9. Descripción textual del CUN_ Instalar recurso.....	43
Tabla 10. Actores del Sistema	49
Tabla 11. Descripción del CUS_ Gestionar Usuario	51
Tabla 12. Descripción del CUS_ Verificar Usuario	51
Tabla 13. Descripción del CUS_ Asignar Rol.....	52
Tabla 14. Descripción del CUS_ Construir Perfil del Usuario	53
Tabla 15. Descripción del CUS_ Asignar Permisos.....	54
Fig. 1 Modelo del Negocio.....	36
Fig. 2. Diagrama de CU de sistema del Módulo de Administración	48

Índice

Introducción	4
<i>CAPÍTULO 1: “Fundamentación Teórica”</i>	8
1.1 Introducción	8
1.2 Conceptos asociados al dominio del problema	8
1.3 Objeto de Estudio	10
1.3.1 Descripción General	10
1.3.2 Técnicas de Autenticación	10
1.3.3 Tipos de ataques informáticos	16
1.3.4 Medidas de Seguridad	19
1.4 Descripción actual del dominio del problema	20
1.5 Situación Problemática	20
1.6 Análisis de otras soluciones existentes	22
1.7 Conclusiones	24
<i>CAPÍTULO 2: “Tendencias y tecnologías actuales a desarrollar”</i>	25
2.1 Introducción	25
2.2 El lenguaje de programación	25
2.3 Metodología	26
2.3.1 Algunas Metodologías	27
2.4 Fundamentación de la Metodología utilizada. Proceso Unificado de Desarrollo	28
2.5 Lenguaje para el Modelado de Objetos	29
2.5.1 Lenguaje Unificado de Desarrollo	29
2.6 Herramientas a utilizar	30
2.6.1 Herramienta de Modelado	30
2.6.1.1 Rational Rose Enterprise Edition	30

2.6.2 Herramientas de desarrollo	31
2.6.2.1 Sistema Gestor de Base de Datos MySQL	31
2.7 Conclusiones	33
<i>CAPÍTULO 3: "Presentación de la solución propuesta"</i>	34
3.1 Introducción	34
3.2 Modelo de Negocio	34
3.2.1 Actores y trabajadores del negocio	34
3.2.2 Diagrama de Casos de Uso del Negocio	35
3.2.3 Descripción textual de los Casos de Uso de Negocio	37
3.2.4 Diagramas de actividades	43
3.2.5 Modelo de Objetos	43
3.3 Especificación de los requisitos del software	43
3.3.1 Requisitos Funcionales del Sistema	43
3.3.2 Requerimientos no Funcionales del Sistema	46
3.4 Descripción del Sistema Propuesto	48
3.4.1 Descripción de los actores	49
3.4.2 Casos de Uso del Sistema	49
3.5 Conclusiones	54
<i>CAPÍTULO 4: "Construcción de la solución propuesta"</i>	55
4.1 Introducción	55
4.2. Arquitectura a implementar	55
4.3 Arquitectura en capas	55
4.3.1 Arquitectura en tres capas	56
4.4 Patrones de diseño	57
4.5 Análisis	59

4.5.1 Diagrama de Clases del Análisis	59
4.6 Diseño	60
4.6.1 Diagrama de Clases del Diseño	60
4.7 Modelo de Despliegue	60
4.8 Conclusiones	61
<i>Conclusiones Generales</i>	62
<i>Recomendaciones</i>	63
<i>Referencias Bibliográficas y Webgrafía</i>	64
<i>Bibliografía Consultada</i>	66
<i>Glosario de Términos</i>	67
<i>Anexos</i>	70

Introducción

Desde las últimas décadas, la tecnología ha adquirido un papel protagónico dentro del desarrollo apresurado de las sociedades. Uno de los principales detonantes de este esparcimiento tecnológico ha sido la adaptación de esta a los diferentes escenarios de las actividades humanas, tanto así, que se puede decir también que las Tecnologías de la Información y las Comunicaciones (TICs) son una nueva forma de comunicación, que cada vez hablan más un lenguaje digital universal, que integran globalmente la producción y distribución de palabras, sonidos e imágenes de nuestra cultura y acomodándolas a los gustos de las identidades y temperamentos de las personas. Las redes informáticas interactivas crecen de modo acelerado, creando nuevas formas y canales de comunicación.

Asimismo, el impacto de las tecnologías de información y su repercusión en el comportamiento de las sociedades, ha obligado a estas últimas a buscar los medios más adecuados para llevar a obtener un mejor provecho y la educación no es una excepción de ello. Esas tecnologías se presentan cada vez más como una necesidad en el contexto de sociedad donde los rápidos cambios, el aumento de los conocimientos y las demandas de una educación de alto nivel constantemente actualizada se convierten en una exigencia permanente.

Cuba no ha estado ajena y se ha dado a la tarea de aplicar las tecnologías en el campo educacional y más específico, en el autoaprendizaje de las lenguas extranjeras, en la que ha puesto todos los esfuerzos en función de informatizar la sociedad. Se han creado instituciones en el país como la Universidad de las Ciencias Informáticas (UCI) que contribuyen al crecimiento informático, y como parte de los aportes que cada día ayudan a elevar el aprendizaje de las lenguajes extranjeras se encuentra la concepción de los Centros de Autoaprendizaje y Servicios de Idiomas Extranjeros (CASIE).

La UCI es uno de los programas con que cuenta el país para alcanzar un logro superior al existente en la rama de la informática, hasta el punto de situarse hoy en día en un lugar destacado, del cual se esperan resultados cada vez más crecientes que contribuyan a conquistar una posición en el mercado mundial. En la UCI se han creado los CASIE, con los cuales se intenta promover el aprendizaje autónomo y facilitar materiales de apoyo a las asignaturas que integran la disciplina Idioma Extranjero.

Los Centros de Autoaprendizaje en la UCI brindan a los estudiantes un ambiente alternativo de aprendizaje, que les permite trabajar de manera independiente en sus debilidades y profundizar en los temas que son de su interés, mediante un entorno que se adapta a su particular estilo de aprender. Estos CASIE trabajan en dos modalidades, los espacios físicos que se encuentran ubicados en los diferentes docentes de la universidad y el espacio virtual o Centro de Autoaprendizaje Virtual de Idiomas Extranjeros (CAVIE), al cual se accede a través de la Intranet de la universidad.

A pesar que se tienen creada en los CASIE toda una serie de condiciones para el autoaprendizaje, aún no se satisfacen todas las necesidades del modelo pedagógico utilizado en la disciplina de Idiomas Extranjeros, por ejemplo, existe un gran déficit de profesores de idiomas, no existe un sistema informatizado en línea las 24 horas, donde el usuario pueda completar una ruta de aprendizaje sin la imprescindible intervención de un profesor, y tampoco existe un sistema que sea capaz de llevar el procesamiento de la información a partir de encuestas respondidas por el usuario y que determine el estado actual del mismo así como su estilo o estilos de aprendizaje.

El proyecto CEntro Virtual de Autoaprendizaje de Lenguas Extranjeras (CEVALE) surge como respuesta a las necesidades planteadas anteriormente de informatizar todos estos procesos que se llevan a cabo, de una forma u otra, en los CASIE, expresándose así, en el Departamento Central de Idiomas Extranjeros (DDC Idiomas) en la UCI. Está concebido como sistema inteligente para que sea capaz de generar rutas de aprendizaje de idiomas. Se espera que este sistema sea montado sobre una plataforma y que genere tantas actividades como tantos recursos existan en este.

Por lo general, los sistemas que se implementan hoy en día, surgen para resolver problemas de alta complejidad y difícilmente estos sistemas no contengan información sensible e incluso, información clasificada, por lo que se hace necesario controlar de alguna manera los accesos, garantizando que sólo sea actualizado por quien se autorice, evitando que personas ajenas a la misma puedan alterar la integridad y consistencia de los datos que pueden ser confidenciales.

Por tanto, se ha identificado como **problema a resolver** el mostrado a continuación:

¿Cómo garantizar la seguridad y controlar el acceso a los procesos de Gestión del Centro de Auto Aprendizaje de Lenguas Extranjeras en la UCI?

Como **objeto de estudio** para este trabajo se tiene *la seguridad en los sistemas de gestión*.

El **Objetivo General** es: *Diseñar el módulo de administración del sistema CEVALE.*

El **campo de acción** determinado es: *las diferentes técnicas de seguridad que se aplican en los sistemas de gestión.*

La **Idea a Defender** de este trabajo será:

Con la obtención de un módulo de administración se lograría la protección de la información en el sistema de gestión de CEVALE.

A los objetivos antes propuestos, para darle cumplimiento se deben desarrollar las siguientes tareas de investigación:

- ✓ Definir el Diseño teórico de la investigación.
- ✓ Definir el Diseño metodológico de la investigación.
- ✓ Caracterizar los diferentes tipos de administraciones en aplicaciones de autoaprendizaje.
- ✓ Estudiar los aspectos relacionados con la seguridad en las aplicaciones Web.
- ✓ Actualizar el conocimiento en torno a la seguridad informática en la UCI, investigando en las técnicas de autenticación, tipos de ataques informáticos y medidas de seguridad a través de entrevista a especialistas de la universidad.
- ✓ Seleccionar las herramientas a utilizar para el diseño del módulo.
- ✓ Seleccionar una metodología de desarrollo de software para la implementación del módulo de administración de CEVALE.
- ✓ Diseñar el módulo de administración de CEVALE.

En el desarrollo de este trabajo se aplicarán los siguientes métodos generales:

Métodos Teóricos:

Analítico y Sintético: Este método tiene como objetivo analizar las teorías y documentos permitiendo la extracción de los elementos significativos que se relacionan con los conceptos asociados al objeto de estudio de la investigación y a las buenas prácticas de la seguridad de las aplicaciones Web.

Histórico-Lógico: Este método será de gran utilidad pues se hará un estudio acerca de la evolución de los Entornos de Aprendizaje en la UCI.

Métodos Empíricos:

Observación: Se hace necesario realizar visitas al Centro de Autoaprendizaje de Lenguas Extranjeras con el fin de conocer cómo se realiza el registro de los datos de los usuarios y la aplicación de los diagnósticos, con el fin de obtener un mayor entendimiento del problema.

Entrevistas: Se realizan entrevistas con el fin de precisar el problema a resolver, así como para la validación de la propuesta que se presenta y la captura de los requisitos.

Y como método particular de la informática se utilizará la:

Modelación: Este método permite la modelación de los diferentes diagramas como: el diagrama del negocio, sistema, análisis, diseño e implementación, según el método de Kruchten, Booch, Rumbaugh, Jacobson, y el proceso Objectory.

La población que será objeto de entrevistas estará compuesta por los once especialistas de seguridad que laboran en la Universidad de las Ciencias Informáticas, los veintidós especialistas del Departamento Central de Idiomas Extranjeros (DDC Idiomas) de la UCI y los nueve integrantes que conforman los otros módulos del sistema. Se toma el 15% como grado de confiabilidad, por tanto la muestra es heterogénea y será de seis individuos por área.

La técnica de muestreo utilizada es **probabilística** y se toma la del **muestreo estratificado** porque subdivide una población heterogénea en una serie de sub-poblaciones homogéneas para garantizar que todas las características de la población heterogénea estén representadas en la muestra.

CAPÍTULO 1: “Fundamentación Teórica”

1.1 Introducción

En este capítulo se hace una descripción de los conceptos que son utilizados en la investigación y que pueden resultar difíciles de comprender, además de exponer de forma exhaustiva el objeto de estudio con el fin de lograr un mayor conocimiento del problema a resolver, con el apoyo de las componentes de las diferentes áreas del conocimiento en el tema de la seguridad en los sistemas de gestión.

Se expone una valoración del estado del arte de la investigación realizada para acercarnos a la frontera del conocimiento referente a los diferentes tipos de autenticación, tipos de ataques informáticos y las medidas de seguridad, además se realiza el análisis de algunas soluciones existentes referentes al entorno de la investigación.

1.2 Conceptos asociados al dominio del problema

Por lo general, al desarrollar sistemas informáticos complejos, se debe tener en cuenta una gran cantidad de elementos substanciales como por ejemplo, la seguridad del sistema para así poder proporcionar un entorno seguro donde el usuario se sienta cómodo a la hora de trabajar, por ello, se requiere la comprensión de algunos conceptos fundamentales sobre la seguridad:

Control de acceso: Es el proceso de conceder permisos a usuarios de acceder a objetos, información, datos como ficheros o impresoras en la red. Está basado en tres conceptos fundamentales: identificación, autenticación y autorización. Incluye autenticar la identidad de los usuarios y autorizar el acceso a datos o recursos. Son necesarios para proteger la confidencialidad, integridad y disponibilidad de los objetos, y por extensión, de la información que contienen, pues permiten que los usuarios autorizados accedan solo a los recursos que ellos requieren para realizar sus tareas.(1)

Identificación: Es la acción por parte de un usuario de presentar su identidad a un sistema, usualmente se usa un identificador de usuario. Establece además, que el usuario es responsable de las acciones que lleve a cabo en el sistema. Esto está relacionado con los registros de auditorías que permiten guardar las acciones realizadas dentro del sistema y rastrearlas hasta el usuario autenticado.

Autenticación: Es el proceso de descubrir y comprobar la identidad de un usuario mediante el examen de sus credenciales y la validación de las mismas consultando a una autoridad determinada. Actualmente se utilizan una gran variedad de mecanismos de autenticación.(1)

Autorización: Es el proceso que determina si el usuario o proceso previamente identificado y autenticado tiene permitido el acceso a los recursos.

Personalización: Es un proceso en el cual la aplicación asume la identidad del invocador y que, en su nombre, realice solicitudes a otros recursos. El acceso se concede o deniega en función de la identidad que se adopte. Si esta identidad dispone de permisos para un recurso, la aplicación que la adopta también dispondrá de los mismos permisos de acceso.

Auditoría: Es un registro cronológico de los eventos relevantes a la seguridad de un sistema. Este registro puede luego examinarse para reconstruir un escenario en particular.

Confidencialidad: La información o los activos informáticos son accedidos solo por las personas autorizadas. Sería fácil mantener la confidencialidad de un sistema si nadie tuviera acceso a él, pero este tendría una disponibilidad nula.

Integridad: La información que está en correspondencia con la realidad solo puede ser modificada por las personas autorizadas y de la forma autorizada.(1)

Disponibilidad: La información solo puede ser accedida por las personas autorizadas en el momento requerido.

Sistema informático: Sistema de procesamiento de la información basado en ordenadores, y personal que realizan funciones de entrada y control con el fin de llevar a cabo una secuencia de operaciones con datos.

1.3 Objeto de Estudio

1.3.1 Descripción General

En los últimos años, con el desarrollo de la informática, se ha tenido un aumento acelerado del nivel de interconexión entre las computadoras, esto conlleva a una serie de ventajas, pero nos obliga a prestar especial atención a los temas relacionados con la seguridad.

El tema de la seguridad hoy en día es un poco difícil, pues por estudios realizados, se ha llegado a la conclusión que de no implantarse medidas de seguridad adecuadas en poco tiempo, se podría frenar el desarrollo de los negocios en Internet, así como influir en el crecimiento de la actividad económica en la red; debido a que los sistemas de información en línea y de realización de transacciones económicas a través de medios telemáticos se enfrentan a numerosos riesgos.

Los ataques de piratas a través de la red son más frecuentes y sus efectos más devastadores, por tal motivo, se hace de vital importancia tomar las medidas adecuadas para evitar fugas o pérdidas de datos, ya sean accidentales o intencionados, en todos los niveles de desarrollo, implantación y mantenimiento de los sistemas informáticos.

1.3.2 Técnicas de Autenticación

En el proceso actual se distinguen cuatro formas de técnicas de autenticación:

- ✚ Sistemas basados en objetos de seguridad.
- ✚ Sistemas de autenticación biométrica.
- ✚ Sistemas de autenticación por certificados.
- ✚ Sistemas basados en contraseñas.

A continuación se describe cada una de ellas:

Sistemas basados en objetos de seguridad.

Distinto a una contraseña, un objeto de seguridad es un objeto físico, o sea, un pequeño dispositivo de hardware que los usuarios cargan consigo para autorizar el acceso a un servicio de red. El dispositivo

puede ser en forma de una tarjeta inteligente o puede estar incorporado en un objeto utilizado comúnmente, como un llavero.(2)

Existe más de una clase de objeto de autenticación, están los bien conocidos generadores de contraseñas dinámicas y los que comúnmente se denominan objetos USB, los cuales no solo permiten almacenar contraseñas y certificados digitales, sino que permiten llevar la identidad digital de la persona.

Los objetos de autenticación proveen un nivel de seguridad adicional utilizando el método conocido como autenticación de dos factores: el usuario tiene un número de identificación personal, que le autoriza como el propietario del dispositivo; luego el dispositivo despliega un número que identifica en forma única al usuario ante el servicio, permitiéndole ingresar.(2)

Las ventajas de utilizar tarjetas inteligentes como medio para autenticar usuarios son muchas frente a las desventajas; se trata de un modelo ampliamente aceptado entre los usuarios, rápido, y que incorpora hardware de alta seguridad tanto para almacenar datos como para realizar funciones de cifrado.

Como principal inconveniente de las tarjetas inteligentes se puede citar el coste adicional que supone para una organización el comprar y configurar la infraestructura de dispositivos lectores y las propias tarjetas; aparte, que un usuario pierda su tarjeta es bastante fácil, y durante el tiempo que no disponga de ella o no puede acceder al sistema, o establecer reglas especiales que pueden comprometer la seguridad (y por supuesto se ha de marcar como tarjeta inválida en una base de datos central, para que un potencial atacante no pueda utilizarla). También la distancia lógica entre la tarjeta de inteligente y su poseedor, simplemente se puede fijar en que la tarjeta no tiene un interfaz para el usuario y puede ser fuente de varios problemas de seguridad.(2)

Sistemas de autenticación biométrica

Parece que en un futuro no muy lejano estos serán los sistemas que se van a imponer en la mayoría de las situaciones en las que se haga necesario autenticar un usuario. Estos sistemas son más amigables para el usuario, no se necesita recordar contraseñas o números de identificación complejos, y como se suele decir, el usuario puede olvidar una tarjeta de identificación en casa, pero nunca olvidará una parte de su cuerpo. Son mucho más difíciles de falsificar que una simple contraseña o una

tarjeta magnética; las principales razones por la que no se han impuesto ya, es su elevado precio, fuera del alcance de muchas organizaciones, y su dificultad de mantenimiento.

Los sistemas antes mencionados son los denominados biométricos, que se basan en rasgos personales distintivos con capacidad de identificar a una persona. Se clasifican en:(2)

- *Fisiológicos*: huella dactilar, iris, retina, cara, geometría de la mano, huella palmar, estructura de las venas, estructura de la oreja, termografía facial.
- *Conductuales*: voz, escritura, firma manuscrita, modo de teclear, modo de andar.

Los sistemas biométricos como ventajas tienen que son:

- Intransferibles: El atacante no los puede utilizar aunque los conozca. Esta característica es suficiente para considerar el sistema mejor que los de contraseña o posesión de objetos.
- No necesitan gestión del usuario, como cambiarlos a menudo, recordar frases largas, guardar objetos.
- Sirven tanto para accesos físicos como lógicos.
- Son muy seguros a cualquier ataque.

Actualmente aún tienen las desventajas de:

- Necesitar electrónica adicional para realizar las lecturas de imágenes y, por lo tanto, son más caros.
- La tecnología no está muy avanzada.
- Tienen un cierto rechazo del usuario delante de la exposición física a un sensor.
- Hay algún prejuicio moral porque las características físicas de las personas son invariables y hacerlas públicas implica estar fichado para toda la vida.
- No son exactos.

Sistemas de autenticación por certificados.

Los sistemas de autenticación por certificados además de autenticar al servidor frente a los clientes, también es posible que los clientes se identifiquen mediante la presentación de un certificado. Los certificados de cliente normalmente contienen información como su nombre, empresa, departamento,

dirección de correo, ciudad, país, etc., lo que permite sofisticados esquemas de control de acceso basándose en uno de estos atributos o en una combinación de varios o en el conjunto del certificado.(3)

De esta manera, el usuario simplemente instala el certificado en su navegador y posteriormente, cuando se conecte al servidor, sólo tiene que presentarlo para que se produzca la autenticación. Los certificados pueden almacenarse localmente en el disco duro del ordenador del usuario o en una tarjeta inteligente. En ambos casos, están además protegidos por una contraseña, para evitar el acceso fraudulento a los mismos.

Presentan las siguientes ventajas:

- Permiten autenticarse en muchos servidores distintos, sin necesidad de recordar multitud de contraseñas, ni lo que es peor, utilizar la misma en todos los servidores.
- Son fáciles de escalar cuando crece el número de usuarios.
- Permiten descentralizar la verificación de permisos de acceso, basándose en la información contenida en el propio certificado.

Presentan como inconvenientes:(4)

Cuando un certificado pierde validez por el motivo que sea, bien porque su clave privada ha sido comprometida, porque ha expirado, porque se ha comprometido la clave de la autoridad que lo certificó, debe añadirse a una lista que contiene todos los certificados que han sido inhabilitados o revocados. Estas listas se conocen como Listas de Revocación de Certificados, y deberían ser consultadas por el servidor cada vez que se le presenta un certificado para ser verificado. Se hace evidente la dificultad de mantener las listas y sincronizar su información, especialmente cuando se trata con un número muy elevado de usuarios. Además, la necesidad de consultarlas en cada autenticación impone una importante sobrecarga de procesamiento que puede llegar a dilatar notablemente el rendimiento del servidor.

Los usuarios suelen ser el mayor obstáculo para que el sistema funcione correctamente. A menudo cometen fallos que hace difícil de gestionar, como que olvidan la clave que protege su certificado, por lo que no pueden acceder al mismo y deben solicitar uno nuevo. Las consecuencias son que no puede

descifrar correos o ficheros que hayan sido cifrados con su clave pública, por lo que han quedado irremisiblemente perdidos, y además se debe añadir a la lista de revocación de certificados.

Dado que el certificado no es más que un fichero protegido por una contraseña, nada impide que lo compartan con otros usuarios, junto con su clave secreta. De ahí la conveniencia de almacenarlos en tarjetas inteligentes, que vuelven más difícil su uso compartido. Almacenar los certificados en el disco duro no es una buena idea.

Sistemas basados en contraseñas.

Una contraseña o clave es una forma de autenticación que utiliza información que solamente el individuo conoce, o sea secreta, para controlar el acceso hacia algún recurso protegido. Está compuesta por un código alfanumérico y en ocasiones solamente numérico. Mientras que las contraseñas crean una seguridad contra los usuarios no autorizados, el sistema de seguridad sólo puede confirmar que la contraseña es válida, y no si el usuario está autorizado a utilizar esa contraseña. Esa es la razón por la que las contraseñas normalmente deben mantenerse en secreto ante aquellos a quien no se le permite el acceso.(2)

Como en todas las técnicas de autenticación, existe una relación entre seguridad y conveniencia (facilidad de uso). Es decir, si algún recurso está protegido por una contraseña, entonces la seguridad se incrementa con la consecuente pérdida de conveniencia para los usuarios. No existe un método que sea el mejor para definir un balance apropiado entre seguridad y conveniencia.

La seguridad de las contraseñas se ve afectada por diversos factores que se mencionarán a continuación:(2)

- *Fortaleza de la contraseña:* Deben ser largas, normalmente más de 7 caracteres, y se deben usar combinaciones de letras mayúsculas y minúsculas, números y símbolos. Ejemplos de contraseñas fuertes serían las siguientes: tastY=wheeT34, pArtei@34! y #23kLLflux.
- *Formas de almacenar las contraseñas:* Se debe usar un algoritmo criptográfico irreversible (o función resumen), los más comunes son SHA1 y SHA2.
- *Método de retransmisión de la contraseña:* Deben ser transmitidas mediante algún método criptográfico, en el caso de las redes locales se usa con mucha frecuencia Kerberos.
- *Longevidad de la contraseña:* Deben ser cambiadas cada 60 días.

Algunos de sus inconvenientes son:(5)

- Los usuarios no suelen elegir contraseñas especialmente robustas, problema que constituye una de las mayores amenazas a la seguridad de un servidor, cuando se trata de contraseñas de una cuenta, o a la confidencialidad de los datos, cuando se trata de un acceso a web.
- Los usuarios pueden revelar sus contraseñas a otras personas, que podrían así, acceder también a los servicios protegidos.
- Los usuarios suelen perder u olvidar sus contraseñas, requiriendo ayuda por parte de los administradores para resolver el problema.

Se llega a la conclusión de utilizar los sistemas basados en contraseña pues las ventajas de utilizar este tipo de autenticación es que es sencillo, barato, y está altamente probado. Es sencillo porque el usuario no tiene que utilizar ningún dispositivo externo para autenticarse en el sistema. Es barato debido a que no se necesita de un coste adicional a la hora de entrar a un sistema y poner tu contraseña, pero en cambio, los sistemas que utilizan objetos para poder autenticarse, necesitan de un gasto extra para utilizar un identificador de objetos. Los sistemas biométricos son mucho más novedosos y se están desarrollando a gran velocidad, aunque tiene desventajas que retardan su desarrollo como el precio de los equipos de captación, conceptos éticos y poca costumbre de utilización. Está probado que los sistemas basados en contraseñas son los más utilizados en el mundo, porque con diferentes variantes son aplicados en casi todos los aspectos de la seguridad de la información.

Igualmente todos los sistemas se pueden combinar para aumentar la seguridad, especialmente el uso de contraseñas normalmente acompaña a los sistemas biométricos y los de objetos. Por ejemplo, el uso de objeto es factible para controles de acceso físico, y se integra fácilmente con otros mecanismos de autenticación como las contraseñas; y en caso de desear bloquear el acceso de un usuario, no tenemos más que retener su tarjeta cuando la introduzca en el lector o marcarla como inválida en una base de datos.

En la Universidad de las Ciencias Informáticas (UCI) se utilizan como técnicas de autenticación hasta el momento, los sistemas Windows contra el Directorio Activo y las Aplicaciones Linux mediante LDAP¹. La más utilizada es la autenticación mediante LDAP debido a que hay muchas más

¹ LDAP: Protocolo de Acceso a Directorios Ligeros.

aplicaciones sobre Linux. En cuanto a la seguridad de estas dos técnicas, en la autenticación de las aplicaciones Windows contra el directorio activo se usa un algoritmo irreversible llamado NTLM² que es bastante seguro mientras que la autenticación de los sistemas Linux con LDAP es algo insegura pues la contraseña es enviada por texto plano a los servidores. En la UCI se estableció por política que la contraseña tuviera como mínimo siete caracteres y en cuanto a su composición no se forzó a que tuviera alguna complejidad para así evitar problemas de olvido de contraseña. Además, se estableció que la contraseña tenga un tiempo de validez de 60 días pasado este tiempo automáticamente la cuenta queda deshabilitada hasta que el usuario cambie la misma.

El método de encriptación usado es el NTLM en los sistemas Windows 2000 y superiores y LM³ en los anteriores. El más recomendado es el NTLM por ser este un algoritmo basado en Hashes e irreversible.

1.3.3 Tipos de ataques informáticos

Las contraseñas son un punto muy vulnerable de la seguridad del sistema de información, si el atacante consigue esa secuencia de pocos caracteres que forma la contraseña tiene la puerta abierta a atacar cualquier recurso. Las formas de poder descubrir las contraseñas de los usuarios se pueden agrupar en:(6)

Con acceso al fichero: Si se tiene acceso al fichero de contraseñas o bases de datos adivinarlas es sólo cuestión de tiempo. Para ello se utilizan programas denominados Crackers que prueban todas las posibilidades hasta encontrar una que al encriptarse coincide. Hay dos métodos de elegir las posibles palabras:

- *Prueba y ensayo o Fuerza bruta:* Se prueban todas las combinaciones de letras, números y signos posibles. Este método es mucho más lento pero al final siempre da resultado (puede tardar meses). Normalmente se va aumentando el número de caracteres de forma progresiva, así se encuentran primero las contraseñas más cortas.
- *Diccionario:* Prueban todas las palabras que pueden aparecer en una enciclopedia, o sea, nombres comunes (de un diccionario), nombres de persona, de animal, geográficos, fechas,

² NTLM: NT Lan Manager.

³ LM: Lan Manager.

números, etc. Esto se puede hacer consecutivamente para varios idiomas y, además, ir haciendo pasadas intercalando números y signos de puntuación. Para que una contraseña sea fácil de recordar debe ser perceptible para el usuario, por lo tanto, ser alguna palabra con significado.

Inyección de código SQL: La inyección SQL consiste en la modificación del comportamiento de nuestras consultas mediante la introducción de parámetros no deseados en los campos a los que tiene acceso el usuario. Este tipo de errores puede permitir a usuarios malintencionados acceder a datos a los que de otro modo no tendrían acceso y, en el peor de los casos, modificar el comportamiento de nuestras aplicaciones.(7)

*Sniffing*⁴: Tiene como objetivo la pasiva interceptación (sin modificación) del tráfico de red. Esto se realiza con sniffers, los cuales son programas que monitorean los paquetes que circulan por la red. Los sniffers pueden ser colocados tanto en una estación de trabajo conectada a la red, como a un equipo de interconexión como un router.

*IP*⁵ *Spoofing*⁶: El objetivo de este ataque es la suplantación de identidad, en este caso la suplantación de la dirección IP. Consiste en generar paquetes de información, con una dirección IP falsa en el encabezado de los paquetes IP. Es una técnica empleada para enmascarar el origen del ataque ya que si el ordenador atacado y el administrador de la red registran esos paquetes donde aparece el IP origen, la persona culpada será otra y no el intruso, incluso se pensará que el ataque proviene de una red distinta a la original, dependiendo de la dirección falsa utilizada.

Ingeniera social: Es la manipulación de las personas para convencerlas de que ejecuten acciones o actos que revelan todo lo necesario para superar las barreras de seguridad. Se plantea que el eslabón más débil de la seguridad es el factor humano. Si el atacante tiene la experiencia suficiente puede engañar fácilmente a un usuario que desconoce las mínimas medidas de seguridad. Esta técnica es una de las más usadas y efectivas a la hora de averiguar nombres de usuarios y contraseña, por

⁴ Sniffing: Programa o dispositivo capaz de leer los datos transmitidos por una red.

⁵ IP: Internet Protocol.

⁶ Spoofing: Suplantar la dirección IP de otro sistema.

ejemplo, puede llamarse a un usuario haciéndose pasar por administrador de la red y requerirle la contraseña con alguna excusa convincente.(8)

*Phishing*⁷: Es un término empleado para distinguir un tipo de ingeniería social caracterizado por la adquisición de información confidencial de forma fraudulenta, como puede ser una contraseña o información detallada sobre tarjetas de crédito u otra información bancaria.

Virus: Son programas que se copian automáticamente y que son capaces de reproducirse sin el permiso o el conocimiento del usuario. Los virus pueden intencionadamente destruir datos en la computadora, formatear el disco duro, alterar información o realizar tareas que entorpecen el buen funcionamiento de la computadora.

Caballos de Troya: Es una pieza de software dañino disfrazado de software legítimo. Son conocidos así, porque su mecanismo de acción es similar al utilizado por los griegos para entrar en Troya. Sus autores los introducen en programas generalmente muy utilizados por el dominio público para que sean propagados a través de copias de los mismos que realicen los usuarios. Los troyanos no son capaces de replicarse por sí mismos y generalmente se emplean para espiar, robar información o controlar remotamente la computadora infectada.(8)

*Keylogger*⁸: Es un programa que captura las pulsaciones del teclado en un sistema comprometido y las almacena en un fichero o las envía por la red de datos. Su objetivo es coleccionar información sensible para un atacante como nombres, contraseñas, números de tarjetas de crédito. Es una herramienta de espionaje muy potente, la cual puede estar instalada en el sistema mucho tiempo sin que alguien note su presencia.

*Spyware*⁹: Los programas espía o spyware son aplicaciones que recopilan información sobre una persona u organización sin su conocimiento. La función más común que tienen estos programas es la de recopilar información sobre el usuario y distribuirlo a empresas publicitarias u otras organizaciones interesadas.

⁷ Phishing: es un intento de obtener información ajena.

⁸ Keylogger: registrador de teclas.

⁹ Spyware: Programas diseñados para espiar el comportamiento de los usuarios.

1.3.4 Medidas de Seguridad

Diseñar un sistema seguro es una tarea muy complicada pues la creciente complejidad de los sistemas informáticos, de los que se demandan más movilidad, flexibilidad y conectividad, ha llevado al crecimiento exponencial de las amenazas. Implantar medidas de seguridad para proteger sistemas informáticos conlleva un proceso de análisis de las vulnerabilidades del sistema. La definición de una política de seguridad y su implementación, a través de una serie de medidas y mecanismos de defensa, obliga a subordinar algunas de las ventajas del sistema informático. Se hace obvio que a mayores y más estrictas medidas de seguridad, menos amigable es el sistema.

Estas medidas de seguridad pueden ser de tres tipos fundamentales: físicas, lógicas y administrativas.(8)

Medidas físicas: Se aplican mecanismos para impedir el acceso directo o físico no autorizado al sistema. También protegen al sistema de desastres naturales o condiciones medioambientales adversas. Existen tres factores fundamentales a considerar:

- El acceso físico al sistema por parte de personas no autorizadas.
- Los daños físicos por parte de agentes nocivos o contingencias.
- Las medidas de recuperación en caso de fallo.

Medidas lógicas: Incluye un conjunto de medidas de acceso lógico a los recursos y a la información, garantizando el uso correcto de los mismos. Se refiere más a la protección de la información almacenada. Entre los tipos de controles lógicos que es posible incluir en el sistema de seguridad se destacan los siguientes:

- Implementación de un sistema de control de accesos a los recursos y la información.
- Uso de la criptografía para proteger los datos y las comunicaciones.
- Empleo de cortafuegos para proteger una red local de Internet.
- Realización de copias de respaldo de la información.
- Monitoreo y auditoría del sistema.

Medidas administrativas: Son aquellas que deben ser tomadas por las personas encargadas de definir la política de seguridad de la institución para ponerla en práctica, hacerla viable y vigilar su correcto funcionamiento. Algunas de las medidas administrativas fundamentales a tomar son las siguientes:

- Documentación y publicación de la política de seguridad de la institución y de las medidas tomadas para ponerla en práctica.
- Establecimiento de un plan de formación del personal.

Un ejemplo de medida de seguridad utilizada en la UCI es que las formas de adivinar la contraseña de un usuario es mediante un ataque de fuerza bruta o de diccionario, para los cuales está establecida una política en el dominio que a los 50 intentos fallidos se desactive temporalmente la cuenta por 15 minutos haciendo esto que un ataque sobre un usuario específico dure posiblemente años.

1.4 Descripción actual del dominio del problema

En estos momentos el Centro de Autoaprendizaje y Servicios de Idiomas Extranjeros (CASIE) recibe cada día usuarios que llegan a este centro para el aprendizaje de idiomas extranjeros. De los mismos se recogen sus datos generales y luego se presentan a realizar una encuesta y de acuerdo al resultado obtenido, este llena un contrato donde se plasman sus datos personales, los estilos de aprendizaje, los horarios y la forma de evaluación con el cual va a realizar su aprendizaje, luego, todos estos datos son archivados en un registro de cliente y todas las actividades que son realizadas por el mismo. Además de llevar en el Registro de Consulta de Libro y en el Registro de cliente los recursos que son utilizados por el cliente ya sean de tipo digital o físico.

1.5 Situación Problemática

Los Centros de Autoaprendizaje y Servicios de Idiomas Extranjeros en la UCI se crearon para propiciar el autoaprendizaje a través de las diferentes asignaturas de las disciplinas de idiomas extranjeros y para apoyar el desarrollo de los idiomas extranjeros del claustro de profesores. Además, se pretende que los estudiantes adquieran habilidades comunicativas y lingüísticas.

También se puede decir que los CASIE de la UCI se caracterizan por estar centrado en los estudiantes, el trabajo cooperativo en ambientes interactivos de aprendizaje, la autogestión del aprendizaje, la solidaridad ciberespacial entre otros.

Además, basado en este enfoque se pueden mencionar algunas ventajas:

- ✓ Es un aprendizaje centrado en las necesidades específicas del estudiante.
- ✓ Es un aprendizaje con grado de independencia con relación al espacio, tiempo, e infraestructura disponibles.

Como la computadora no castiga al estudiante por los errores que puede cometer, le facilita revisar sus tareas sin sentirse presionado por la mirada de alguien que espera.

El asesor no responde a la figura tradicional del maestro por lo que su relación con el alumno se da en gran medida a través de las asesorías y las rutas de trabajo que propone. Estas rutas de trabajo deben ser claras y precisas para evitar ambigüedades.

Es lógico decir que se cuenta no solo con el espacio físico sino que también se cuenta con el CAVIE o espacio virtual, el cual fue creado por un grupo de especialistas para satisfacer las necesidades existentes en los CASIE. Este espacio virtual está conformado principalmente por cinco módulos (módulo informativo, módulo de diagnóstico, módulo de asesoría, módulo de recursos y módulo de evaluación) los cuales brindan por sí mismos, algunos servicios que están guiados para dar respuesta a las necesidades existentes. A continuación se muestra de una forma sintética los servicios que actualmente brindan estos módulos.

Módulo informativo: Se brinda toda la información necesaria para convertirse en un usuario de la institución: visita virtual donde se explica toda la estructura del CASIE y los servicios que se ofrecen, donde reciben información acerca de que es autoaprendizaje y autogestión, así como aplicaciones informáticas para el proceso de inscripción o autenticación.

Módulo de diagnóstico: Se aplicarán encuestas computarizadas para conocer las necesidades de los usuarios, sus estilos y estrategias de aprendizaje, el tipo de inteligencia que poseen y exámenes de nivel de desarrollo en los idiomas extranjeros.

Módulo de asesoría: En base a los resultados en los diagnósticos realizados los asesores ofrecerán a los usuarios diferentes rutas de aprendizaje, así como consejos acerca de las mejores estrategias de aprendizaje a utilizar para alcanzar los objetivos. Se negocia con los usuarios y se inicia el diario y portafolio de aprendizaje, que permiten llevar a cada usuario el récord de su proceso de aprendizaje.

Módulo de recursos: A través de aplicaciones informáticas se accede a los recursos didácticos, organizados según catálogos en base a localización de los recursos (físicos en el CASIE, virtuales en la Intranet o virtuales en Internet), idiomas, habilidades, componentes del lenguaje, niveles de desarrollo, objetivos de aprendizaje, siguiendo como estrategia las rutas de aprendizaje diseñadas para cada usuario con el asesor.

Módulo de evaluación: Los usuarios pueden solicitar en cualquier momento la realización de procesos de auto-evaluación en todos los niveles y habilidades, como un proceso que les permitirá la retroalimentación inmediata de su proceso de aprendizaje. Se prestarán además, servicios de exámenes internacionales y procesos de acreditación de competencias en varios idiomas.

Pero aun así, existen en los CASIE, insatisfacciones que no se les ha dado solución como que los profesores de inglés después que se aplica la encuesta, en base a los resultados, tienen que conformar una ruta de aprendizaje al estudiante que esté en unión con sus insuficiencias pedagógicas y entonces el profesor no le puede dedicar mayor tiempo al trabajo de asesoría, además, no existe un espacio que se caracterice por una gran interactividad y que esté abierto al crecimiento continuo, que esté disponible las 24 horas, que permita procesar de forma informatizada la información que se genera al aplicar las encuestas al usuario determinando los estilos de aprendizaje del mismo, que tenga definido e implementado procedimientos de seguridad informática para evitar el robo de información, suplantación de identidades, daños directos a la aplicación y transformación de la integridad y confidencialidad de la información.

1.6 Análisis de otras soluciones existentes

Profesionales de la seguridad hoy en día se enfrentan a la pérdida de datos de aplicaciones web y servicios web de gestión de los riesgos de seguridad. Necesitan un sistema informatizado, programado, dinámico, continuo y de aplicación de análisis de seguridad. Con el objetivo de facilitar recursos que capaciten a los desarrolladores de sistemas informáticos a afrontar dichos retos, y de solucionar los problemas de este tipo en las aplicaciones Web. Un sin número de productos y sistemas se han desarrollado a nivel mundial y aunque no cubren en su totalidad la diversidad de obstáculos de este género que van surgiendo, brindan grandes beneficios en cuestiones de protección de datos y sistemas.

Assessment Management Platform (AMP)

AMP, producto de SPI Dynamics, es la primera y única plataforma de evaluación para abordar plenamente las complejidades de la actual aplicación web, los programas de análisis de seguridad. Establece el estándar para la seguridad mundial, los programas de avanzada que permiten a las distintas partes interesadas y los participantes para obtener la información que necesitan acerca de la seguridad de las aplicaciones y participar en el proceso de evaluación y remediación, al tiempo que se mantiene un control centralizado de la seguridad.

Permite a las organizaciones realizar ilimitado control automático de evaluación de la seguridad mientras que la consolidación de toda la información en un tiempo real, de alto nivel, la vista de escritorio de una empresa de riesgo actual postura, y la política de cumplimiento. Este enfoque consolida y se resumen todos los esfuerzos de aplicación de análisis de seguridad en toda la organización para que se pueda evaluar la seguridad de su organización en la capa de aplicaciones.(9)

DevInspect™

Producto de SPI Dynamics, simplifica la seguridad para desarrolladores de forma automática y la fijación de la búsqueda de vulnerabilidades y aplicación que permite a los desarrolladores construir aplicaciones web seguras y servicios Web de manera rápida y fácil, sin afectar los calendarios o que requieren experiencia en seguridad.

Apoya J2EE para desarrolladores de Java, así como el marco para ASP.NET. Además, DevInspect es la primera y única aplicación de software de seguridad de productos para los desarrolladores con el apoyo directo para la obtención de Microsoft ASP.NET, AJAX (antes codenamed "Atlas") y otras tecnologías AJAX.(10)

Central Authentication Service (CAS)

CAS es un código abierto de servicio de autenticación central construida por y para la comunidad de la educación superior y utilizada por cientos de organizaciones educativas y comercial internacional. Con origen en la Universidad de Yale y patrocinada por mayor y JA-SIG desde 1994, el CAS es ampliamente desplegado para proporcionar a la empresa escala, un solo signo de confianza en una serie de aplicaciones (incluyendo uPortal, Sakai, TikiWiki, MULE, Moodle, y muchos otros) a través de

la comunidad el apoyo CAS bibliotecas cliente. Los usuarios se autentican al servidor CAS y solo necesitan hacerlo una sola vez por sesión del navegador.(11)

Moodle (Entorno Dinámico de Aprendizaje Modular Orientado a Objetos)

Moodle es un sistema de gestión de cursos de libre distribución que en términos de arquitectura es una aplicación web, conformada por más de trescientos módulos, donde el administrador es el único que tiene potestad para crear cursos o dar permisos a los que considere oportuno para que lo hagan. La gestión de usuarios se realiza de forma centralizada. Cada usuario, alumno o profesor tendrá acceso a las asignaturas que le corresponden sin que tenga que hacer mayores gestiones. Soporta un rango de mecanismos de autenticación a través de módulos de autenticación, que permiten una integración sencilla con los sistemas existentes. Además, se ha puesto énfasis en una seguridad sólida en toda la plataforma. Todos los formularios son revisados, las cookies¹⁰ cifradas, etc.

Todas estas entidades han desarrollado buenas herramientas a favor de la seguridad de las aplicaciones, pero cumplen funcionalidades propias que no satisfacen la totalidad de las necesidades de un sistema Web específico, los servicios indispensables no son gratuitos y emplean metodologías de certificación y chequeo que se sitúan en escalones inalcanzables en el mundo de la informática.

1.7 Conclusiones

Con la descripción general del objeto de estudio, y el apoyo de las componentes de las diferentes áreas del conocimiento del tema de la seguridad en los sistemas de gestión, se logró una valoración del estado del arte para así llegar a la conclusión de que el sistema de autenticación más factible a utilizar es el basado en contraseña por sus características y ventajas antes expuestas.

¹⁰ Cookies: pequeños archivos de texto que son descargados automáticamente.

CAPÍTULO 2: “Tendencias y tecnologías actuales a desarrollar”

2.1 Introducción

En este capítulo se detalla todo el entorno tecnológico en el cual se basa y desarrolla la aplicación. Primeramente, se abordará el lenguaje de programación que se utilizará donde se fundamenta la selección de dicho lenguaje utilizando sus principales características. También se aborda el tema de la metodología a utilizar, mencionando y explicando brevemente características de diferentes metodologías, enfatizando en la que se utilizará. Se fundamenta las herramientas utilizadas, tanto la herramienta de modelado como la herramienta de desarrollo, donde se explican características específicas de cada uno.

2.2 El lenguaje de programación

PHP es un lenguaje interpretado de alto nivel impregnado en páginas HTML pero sus instrucciones son ejecutadas del lado del servidor. Como producto de código abierto, PHP goza de la ayuda de un gran grupo de programadores, permitiendo que los fallos de funcionamiento se encuentren y se reparan rápidamente.

El código se pone al día continuamente con mejoras y extensiones del lenguaje para ampliar las capacidades de PHP. Como parte de este desarrollo se cuenta con su última versión PHP5 que tiene como principal objetivo mejorar los mecanismos de la programación orientada a objetos (POO) solucionando así las carencias de las anteriores versiones lo cual es un paso necesario para conseguir que PHP sea un lenguaje apto para todo tipo de aplicaciones y entornos. A continuación se mencionan algunas de las características del trabajo con POO en PHP5:(12)

- ✚ Nombres fijos para los constructores y destructores.
- ✚ Acceso público, privado y protegido a propiedades y métodos.
- ✚ Posibilidad de uso de interfaces.
- ✚ Métodos y clases final.
- ✚ Operador instancia de.

2.2.1 IDE Eclipse

Un IDE es un editor de código que sirve para depurar y facilitar las diferentes tareas necesarias en el desarrollo de cualquier tipo de aplicación.

Eclipse es una comunidad de código abierto, cuyos proyectos se centran en la construcción de una plataforma de desarrollo abierta comprendida por frameworks extensibles, herramientas y versiones para la construcción, desarrollo y software de gestión dentro de su ciclo de vida. La versión actual de Eclipse dispone de las siguientes características:(13)

- ✚ Editor de texto.
- ✚ Resaltado de sintaxis.
- ✚ Compilación en tiempo real.
- ✚ Soporte de las versiones 4 y 5 de PHP, ya sea bien de forma genérica a todos los proyectos que se generen, o bien de forma individual a cada uno con previa especificación en las propiedades del proyecto en cuestión.
- ✚ Gestión y exploración de todas las clases generadas a lo largo de la edición del código o bien que se hayan importado de otra librería de PHP, estas clases las toma Eclipse y pueden ser usadas en todo el proyecto como si fuera parte de la librería estándar de PHP.

El SDK de Eclipse incluye las herramientas de desarrollo de Java, ofreciendo un IDE con un compilador de Java interno y un modelo completo de los archivos fuente de Java. Esto permite técnicas avanzadas de refactorización y análisis de código. El IDE también hace uso de un espacio de trabajo, en este caso un grupo de metadato en un espacio para archivos plano, permitiendo modificaciones externas a los archivos en tanto se refresque el espacio de trabajo correspondiente.

2.3 Metodología

La metodología en el desarrollo de un software se puede definir como un conjunto de pasos y procedimientos que sirven de apoyo para la guía de realizar un software de calidad. Es el conjunto de técnicas, herramientas y soporte documental que ayuda a los desarrolladores a realizar nuevos software con los requisitos requeridos. Por lo tanto representa el camino a seguir para desarrollar software de manera sistemática.

En un proyecto de desarrollo de software la metodología define ¿Quién debe hacer Qué? ¿Cuándo? y ¿Cómo debe hacerlo? Una metodología es un proceso.

2.3.1 Algunas Metodologías

Extreme Programing (XP)

XP es una metodología ágil, centrada en potenciar las relaciones interpersonales como clave para el éxito en desarrollo de software, promoviendo el trabajo en equipo, preocupándose por el aprendizaje de los desarrolladores, y propiciando un buen clima de trabajo. Se basa en realimentación continua entre el cliente y el equipo de desarrollo, comunicación fluida entre todos los participantes, simplicidad en las soluciones implementadas y coraje para enfrentar los cambios. XP se define especialmente para proyectos con requisitos imprecisos y muy cambiantes, y donde existe un alto riesgo técnico.(14)

Feature Driven Development (FDD)

FDD es una metodología ágil diseñada por Peter Coad, Eric Lefebvre y Jeff De Luca. Se basa en un proceso iterativo con iteraciones cortas que producen un software funcional que el cliente y la dirección de la empresa pueden ver y monitorear. Define claramente entregas tangibles y formas de evaluación del progreso del proyecto. Ayuda a contrarrestar situaciones como el exceso en el presupuesto, fallas en el programa o el hecho de entregar menos de lo deseado. Propone tener etapas de cierre cada dos semanas. Se obtienen resultados periódicos y tangibles.(14)

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 tiene las siguientes características:(15)

- ✚ *Adaptable*: Usado en cualquier parte como un mapa, del cual su uso es limitado a un específico lugar.
- ✚ *Escalable*: Puede organizar equipos tan pequeños entre 3 o 4 personas.

- ✚ *Flexible*: Es utilizada en el ambiente de desarrollo de cualquier cliente.
- ✚ *Tecnología Agnóstica*: Porque puede ser usada para desarrollar soluciones basadas sobre cualquier tecnología.

2.4 Fundamentación de la Metodología utilizada. Proceso Unificado de Desarrollo

El Proceso Unificado de Desarrollo utiliza el Lenguaje Unificado de Modelado (UML) como lenguaje de representación visual. Es orientado a objetos. Agrupa las actividades en grupos lógicos definidos en nueve flujos de trabajos principales. Los seis primeros son conocidos como flujos de ingeniería (Modelar el Negocio, Requerimientos, Análisis y Diseño, Implementación y Prueba) y los tres últimos como de apoyo (Gestión de proyecto, Gestión de Configuración y Cambio, Ambiente).

RUP en dos dimensiones representa el proceso en el que se grafican los flujos de trabajo y la fases y muestra la dinámica expresada en iteraciones y puntos de control. Divide en cuatro fases el desarrollo del software (Conceptualización, Elaboración, Construcción y Transición) en cada una se obtiene un producto final y cada una está desarrollada mediante el ciclo de iteraciones, las cuales tienen como función la evaluación de las iteraciones precedentes.

Características de RUP

El ciclo de vida de RUP se caracteriza por tres elementos fundamentales que es lo que define su verdadera importancia y se resume en:

Dirigido por casos de uso: Se basa en los casos de uso que reflejan lo que los usuarios futuros necesitan y desean, lo cual se capta cuando se modela el negocio y se representa a través de los requerimientos. Luego los modelos que se realizan guían todo el proceso, pues estos se obtienen como resultado de avanzar a través de los diferentes flujos de trabajo que se derivan de los casos de uso.

Centrado en la arquitectura: La arquitectura muestra la visión común del sistema completo en la que el equipo de proyecto y los usuarios deben de estar de acuerdo, por lo que describe los elementos del modelo que son importantes para su construcción, los cimientos del sistema que son necesarios como base para comprenderlos, desarrollarlo y producirlo económicamente. RUP se desarrolla mediante iteraciones, comenzando por los casos de uso relevantes desde el punto de vista de la arquitectura.

Iterativo e Incremental: RUP propone que cada fase se desarrolle en iteraciones. Una iteración involucra actividades de todos los flujos de trabajo, aunque desarrolla fundamentalmente algunos más que otros. Las iteraciones hacen referencia a pasos en los flujos de trabajo, y los incrementos, al crecimiento del producto.

Después de analizar las características de cada metodología antes expuestas, se determinó que se utilizará para el desarrollo del sistema la metodología el Proceso Unificado de Desarrollo (RUP).

2.5 Lenguaje para el Modelado de Objetos

El lenguaje de modelado de objetos es un conjunto estandarizado de símbolos y de modos de disponerlos para modelar parte de un diseño de software orientado a objetos. El uso de un lenguaje de modelado es más sencillo que la auténtica programación, pues existen menos medios para verificar efectivamente el funcionamiento adecuado del modelo.(16)

2.5.1 Lenguaje Unificado de Desarrollo

El lenguaje que se utilizará para el modelado de objetos es el Lenguaje Unificado de Modelado (UML) porque es un lenguaje de modelado visual que se usa para especificar, visualizar, construir y documentar artefactos de un sistema de software. Captura decisiones y conocimiento sobre los sistemas que se deben construir. Se usa para entender, diseñar, hojear, configurar, mantener, y controlar la información sobre tales sistemas. Está pensado para usarse con todos los métodos de desarrollo, etapas del ciclo de vida, dominios de aplicación y medios. Está pensado para ser utilizado en herramientas interactivas de modelado visual que tengan generadores de código así como generadores de informes. La especificación de UML no define un proceso estándar pero está pensado para ser útil en un proceso de desarrollo iterativo. Pretende dar apoyo a la mayoría de los procesos de desarrollo orientados a objetos.(16)

UML no es un lenguaje de programación. UML no es un lenguaje altamente formal pensado para probar teoremas. UML es un lenguaje de modelado de propósito general. UML es un lenguaje de modelado discreto. No se creó para modelar sistemas continuos como los basados en ingeniería y física. UML quiere ser un lenguaje de modelado universal, de propósito general, para sistemas discretos, tales como los compuestos por software o lógica digital.

Además se utiliza UML porque es el lenguaje de modelado que por defecto trae definido RUP.

2.6 Herramientas a utilizar

Para desarrollar la propuesta del software se hace necesario el uso de varias herramientas. Estas herramientas facilitan el trabajo, haciendo que este se realice de una forma mucho más eficiente. Para su análisis, las mismas se han dividido en dos grupos: las herramientas de modelado y las herramientas de desarrollo.

2.6.1 Herramienta de Modelado

Las herramientas de modelado de objetos, son muy importante para hacer el análisis del sistema, pues le brinda una mejor precisión al trabajo. Para modelar la aplicación se utilizó Rational Rose Enterprise Edition.

2.6.1.1 Rational Rose Enterprise Edition

Rational Rose es la herramienta CASE desarrollada por los creadores de UML (Booch, Rumbaugh y Jacobson), que cubre todo el ciclo de vida de un proyecto: concepción y formalización del modelo, construcción de los componentes, transición a los usuarios y certificación de las distintas fases y entregables.(16)

Es la herramienta CASE que comercializan los desarrolladores de UML y que soporta de forma completa la especificación del UML.

Esta herramienta propone la utilización de cuatro tipos de modelos para realizar el diseño del sistema, utilizando una vista estática y otra dinámica de los modelos del sistema, uno lógico y otro físico. Permite crear y refinar estas vistas creando de esta forma un modelo completo que representa el dominio del problema y el sistema de software.

Rational Rose permite que haya varias personas trabajando a la vez en el proceso iterativo controlado, para ello posibilita que cada desarrollador opere en un espacio de trabajo privado que contiene el modelo completo y tenga un control exclusivo sobre la propagación de los cambios en ese espacio de trabajo.(16)

También es posible descomponer el modelo en unidades controladas e integrarlas con un sistema para realizar el control de proyectos que permite mantener la integridad de dichas unidades.

Características principales:

- Admite como notaciones: UML, COM, OMT y Booch.
- Realiza Chequeo semántico de los modelos.
- Ingeniería “de ida y vuelta”: Rose permite generar código a partir de modelos y viceversa.
- Desarrollo multiusuario.
- Integración con modelado de datos.
- Generación de documentación.
- Tiene un lenguaje de script para poder ampliar su funcionalidad.
- Disponible en múltiples plataformas.

2.6.2 Herramientas de desarrollo

En toda plataforma Web para lograr un software eficiente se necesitan herramientas que posibiliten el almacenamiento de los datos, es decir, un Gestor de Base de Datos (BD).

2.6.2.1 Sistema Gestor de Base de Datos MySQL

MySQL es un sistema de gestión de bases de datos desarrollado y distribuido por MySQL.AB compañía comercial, fundada por los desarrolladores de MySQL.(17)

MySQL es un sistema de gestión de bases de datos.

Una base de datos es una colección estructurada de datos para añadir, acceder, y procesar los datos almacenados en una base de datos, necesita un sistema de gestión de base de datos como MySQL Server. Al ser los computadores muy buenos en tratar grandes cantidades de datos, los sistemas de gestión de bases de datos juegan un papel como aplicaciones autónomas o como parte de otras aplicaciones.

MySQL es un sistema de gestión de bases de datos relacionales.

Una base de datos relacional almacena datos en tablas separadas en lugar de poner todos los datos en un gran almacén. Esto añade velocidad y flexibilidad. La parte SQL de "MySQL" se refiere a "Structured Query Language". SQL es el lenguaje estandarizado más común para acceder a bases de datos y está definido por el estándar ANSI/ISO SQL.

Principales características de MySQL:(18)

- ✓ *Interioridades y portabilidad.*
- ✓ Probado con un amplio rango de compiladores diferentes.
- ✓ Funciona en diferentes plataformas.
- ✓ Proporciona sistemas de almacenamiento transaccional y no transaccional.
- ✓ Un sistema de reserva de memoria muy rápido basado en hilos.

El servidor está disponible como un programa separado para usar en un entorno de red cliente/servidor. También está disponible como biblioteca y puede ser incrustado en aplicaciones autónomas. Dichas aplicaciones pueden usarse por sí mismas o en entornos donde no hay red disponible.

Seguridad

Un sistema de privilegios y contraseñas que es muy flexible y seguro, y que permite verificación basada en el host. Las contraseñas son seguras porque todo el tráfico de contraseñas está encriptado cuando se conecta con un servidor.

Escalabilidad y límites

Soporte a grandes bases de datos. Se usa MySQL Server con bases de datos que contienen 50 millones de registros.

Localización

El servidor puede proporcionar mensajes de error a los clientes en muchos idiomas.

Todos los datos se guardan en el conjunto de caracteres elegido.

Clientes y herramientas

MySQL server tiene soporte para comandos SQL para chequear, optimizar, y reparar tablas.

2.7 Conclusiones

Después de analizar las características, ventajas y desventajas de cada herramienta que fueron explicadas anteriormente, se llegó a la conclusión que se utilizará las siguientes herramientas:

- ✚ Lenguaje de Programación PHP5.
- ✚ Metodología RUP.
- ✚ Lenguaje de Modelado UML.
- ✚ Rational Rose Enterprise Edition.
- ✚ Gestor de Base de Datos MYSQL.

Para así garantizar la estabilidad del sistema y su compatibilidad entre los otros módulos que también se están desarrollando.

CAPÍTULO 3: “Presentación de la solución propuesta”

3.1 Introducción

En este capítulo se realizará el análisis y representación de los procesos del negocio y continúa con la modelación del sistema propuesto. Primeramente se hace una descripción de los actores y trabajadores del negocio, luego se enumeran los requerimientos del sistema, tanto los funcionales como los no funcionales. Se representa el diagrama de los casos de uso del sistema los cuales muestran la solución propuesta del sistema así como la descripción textual de ellos.

3.2 Modelo de Negocio

Cuando se modela el negocio lo primero que se identifica son los actores y los trabajadores que participan en este. Luego se definen los artefactos que se generan, el vocabulario y las reglas que se emplean en la elaboración de los diagramas de casos de uso del negocio, actividad y de clase del modelo de objetos.

3.2.1 Actores y trabajadores del negocio

Un actor del negocio es cualquier individuo, grupo, entidad, organización, máquina o sistema de información externos; con los que el negocio interactúa. Lo que se modela como actor es el rol que se juega cuando se interactúa con el negocio para beneficiarse de sus resultados. Cada actor del negocio debe definirse brevemente con su responsabilidad y por qué interactúa con el negocio. A continuación se muestra la justificación de los actores y trabajadores del negocio.(19)

Actores del negocio	Justificación
Usuario	Es el actor que se beneficia con los servicios de Autoaprendizaje de Idiomas que brindan los CASIE. Representa un estudiante, un profesor o un trabajador.

Tabla 1. Actores del negocio

Un trabajador del negocio representa a personas, o sistemas (software) dentro del negocio que son los que realizan las actividades que están comprendidas dentro de un caso de uso. Estos trabajadores están dentro de la frontera del negocio, son los que en un futuro se convertirán en usuarios del sistema que se quiere construir. Cada trabajador del negocio debe definirse brevemente con su responsabilidad dentro del negocio.(19)

Trabajadores del negocio	Justificación
Recepcionista	Se encarga de registrar todos los datos de los usuarios que visiten los CASIE.
Asesor	Se encarga de emitir una calificación de las encuestas aplicada a los usuarios, así como orientarlos con las posibles rutas a seguir para su autoaprendizaje.
Técnico General	Se encarga de aplicar las encuestas a los usuarios una vez creado el contrato, así como registrar los datos de cuando se utilice un material para el autoaprendizaje.
Técnico Informático	Se encarga de instalar el software que el usuario necesite para su autoaprendizaje.

Tabla 2.Trabajadores del negocio

3.2.2 Diagrama de Casos de Uso del Negocio

El diagrama de casos de uso del negocio representa la relación que existe entre los casos de uso y los actores que los inicializan. El diagrama que se muestra a continuación muestra los tres casos de uso base del negocio con las inclusiones y extensiones de estos casos de uso.

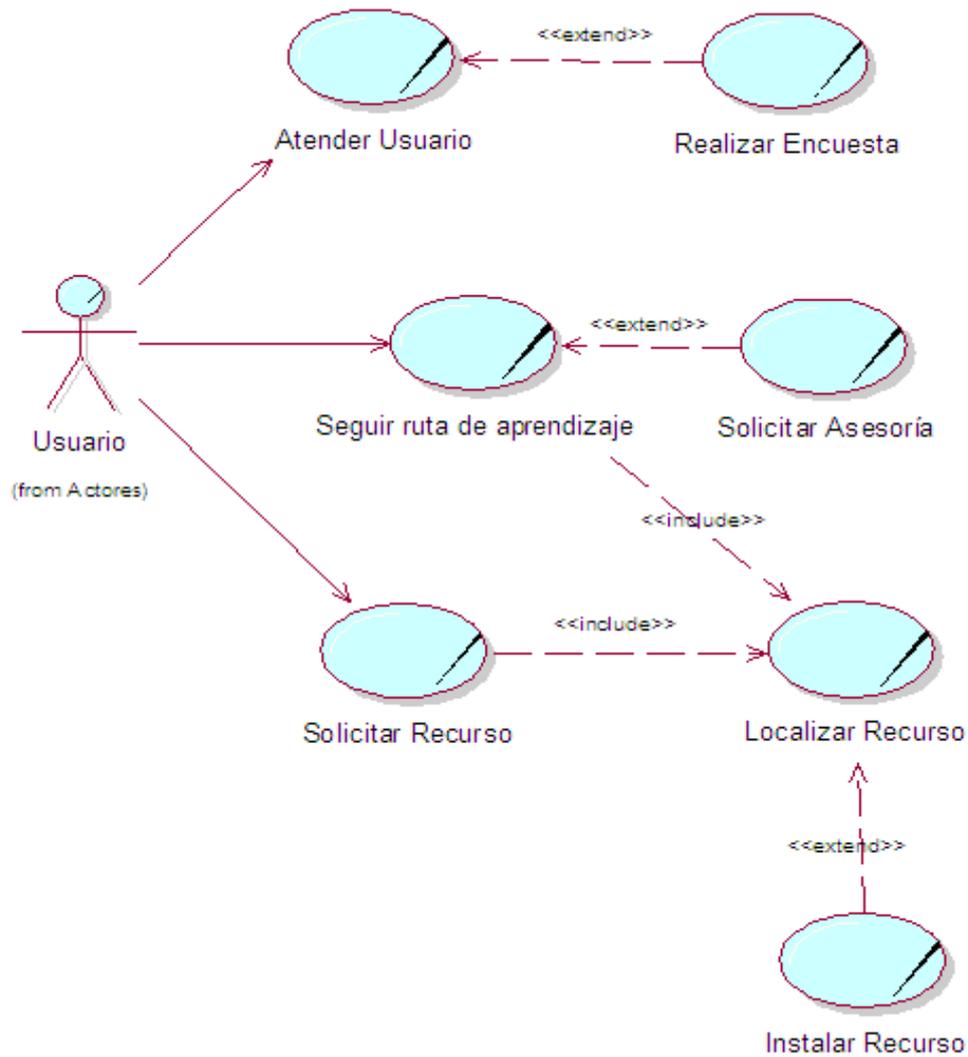


Fig. 1 Modelo del Negocio

Los objetivos de modelar el negocio son:

- ✚ Comprender la estructura y la dinámica de la organización en la cual se va a implantar un sistema.
- ✚ Comprender los problemas actuales de la organización e identificar las mejoras potenciales.
- ✚ Asegurar que los clientes y los desarrolladores tengan un entendimiento común de la organización.

3.2.3 Descripción textual de los Casos de Uso de Negocio

Caso de Uso:	Atender usuario
Actores:	Usuario(inicia)
Trabajadores:	Recepcionista.
Resumen:	El caso de uso se inicia cuando un usuario llega al CASIE y solicita el servicio de este para el Autoaprendizaje.
Precondiciones:	
Flujo Normal de Eventos	
Acción del Actor	Respuesta del Negocio
1-El caso de uso se inicia cuando el usuario llega al CASIE y solicita Auto-aprender un idioma.	2- La recepcionista le da la bienvenida al CASIE y le pide los datos del usuario.
3-El usuario informa los datos.	4-La recepcionista recibe la información del usuario. 5- La recepcionista verifica si es la primera vez que el usuario viene al CASIE 6- La recepcionista registra los datos en el registro de usuarios. 7- Si es la primera vez, la recepcionista le informa al técnico general que le aplique una encuesta al usuario, finalizando así, el caso de uso, sino ver Flujo Alterno 1.
Flujos Alternos 1	
Acción del Actor	Respuesta del Negocio
	1-Sino es la primera vez que el usuario viene al CASIE la recepcionista consulta la fase y nivel del usuario en el libro de categorías. 2- La recepcionista informa al usuario fase y nivel en que se encuentra.
3- El usuario recibe la información, terminando así el caso de uso.	
Pos condiciones	El usuario queda registrado en el registro de cliente.

Capítulo 3: Presentación de la solución propuesta

Prioridad	Crítico.
------------------	----------

Tabla 3.Descripción textual del CUN_ Atender Usuario.

Caso de Uso:	Realizar Encuesta.
Actores:	Usuario
Trabajadores:	Técnico general(inicia), Asesor
Resumen:	El caso de uso se inicia cuando el técnico general entrega la encuesta al usuario, el usuario realiza la encuesta, el asesor califica la encuesta e informa la fase, nivel y la ruta a seguir.
Precondiciones:	Asistir por primera vez en el CASIE.
Flujo Normal de Eventos	
Acción del Actor	Respuesta del Negocio
	1-El técnico general recibe orientación de aplicar encuesta. 2-El técnico general entrega la encuesta al usuario.
3- El usuario recibe la encuesta. 4- El usuario responde la encuesta. 5-El usuario entrega la encuesta al técnico general.	6- El técnico general recibe la encuesta respondida por el usuario. 7- El técnico general entrega la encuesta respondida por el usuario al asesor.
	8- El asesor recibe la encuesta. 9- El asesor califica la encuesta. 10- El asesor registra la fase y el nivel en que se encuentra el usuario en el registro de categorías. 11-El asesor crea un contrato en convenio con el usuario.
12- El usuario firma el contrato. 13- El usuario entrega el contrato.	14- Recoger el contrato. 15- El asesor firma el contrato. 16- El asesor informa al usuario la ruta de aprendizaje a seguir, finalizando así, el caso de uso.
Poscondiciones	Queda confeccionada la ruta de aprendizaje del usuario.
Prioridad	Crítico

Tabla 4.Descripción textual del CUN_Realizar Encuesta

Caso de Uso:	Solicitar Asesoría
Actores:	Usuario (inicia)
Trabajadores:	Asesor
Resumen:	El caso de uso se inicia cuando el usuario solicita la ayuda del asesor, debido a la existencia de una duda en la ruta de Autoaprendizaje trazada. El asesor brinda la asesoría al cliente finalizando el caso de uso.

Precondiciones:	El usuario tiene una ruta de autoaprendizaje trazada.	
Flujo Normal de Eventos		
Acción del Actor	Respuesta del Negocio	
1- El usuario solicita la ayuda del asesor.	2- El asesor recibe el pedido de ayuda por parte del usuario.	
	3- El asesor pregunta al usuario las dudas que este tiene.	
4-El usuario le informa la duda que tiene.	5- El asesor recibe la información de la duda del usuario.	
	6- El asesor analiza la pregunta realizada.	
	7- Si la pregunta puede ser respondida, el asesor emite la respuesta de la pregunta al usuario, sino ir a Flujos Alternos 1.	
8- El usuario recibe la respuesta del asesor a la duda planteada, finalizando así, el caso de uso.		
Flujos Alternos 1		
Acción del Actor	Respuesta del Negocio	
	1-Si la duda no puede ser aclarada el asesor comunica al usuario no estar en condiciones para aclarar la duda.	
2- El usuario recibe la respuesta del asesor de no estar en condiciones de aclarar la duda planteada.		
3- Si el usuario tiene más dudas ir a la actividad 4 del flujo normal de eventos, sino ir a Flujos Alterno 2.		
Flujos Alternos 2		
Acción del Actor	Respuesta del Negocio	
1-Si el usuario no tiene más dudas lo comunica al asesor y continúa con su aprendizaje, finalizando así el caso de uso.		
Poscondiciones	El usuario aclaró o no las dudas existentes en la tarea que estaba realizando.	
Prioridad	Crítico	

Tabla 5.Descripción textual del CUN_Solicitar Asesoría

Caso de Uso:	Seguir ruta de aprendizaje	
Actores:	Usuario(inicia)	
Trabajadores:	Recepcionista	
Resumen:	El caso de uso se inicia cuando el usuario decide seguir una ruta de aprendizaje para lo cual cuenta con una lista de contenidos y recursos la cual debe seguir.	
Precondiciones:	Tiene que existir una ruta de autoaprendizaje y un contrato creado.	
Flujo Normal de Eventos		
Acción del Actor	Respuesta del Negocio	

<p>1- El caso de uso se inicia cuando el usuario llega al CASIE para seguir una ruta de aprendizaje.</p> <p>2- El usuario solicita a la recepcionista ver el contrato para comprobar ruta de aprendizaje.</p>	<p>3- La recepcionista busca el contrato del usuario.</p> <p>4- La recepcionista entrega el contrato al usuario.</p>
<p>5- El usuario comprueba la ruta de aprendizaje.</p> <p>6-El usuario escoge la actividad que está al principio de la lista de actividades no realizadas.</p>	<p>8-La recepcionista recoge el contrato.</p> <p>9- La recepcionista guarda el contrato del usuario, finalizando así el caso de uso</p>
<p>7- El usuario entrega el contrato a la recepcionista.</p>	
Poscondiciones	El usuario escoge las actividades según la ruta de aprendizaje trazada por el asesor.
Prioridad	Crítico

Tabla 6.Descripción textual del CUN_Seguir ruta de aprendizaje

Caso de Uso:	Solicitar recurso.
Actores:	Usuario (inicia)
Trabajadores:	Técnico general.
Resumen:	El caso de uso se inicia cuando un usuario solicita al Técnico General un recurso. El Técnico General recibe la petición y brinda al cliente la localización del recurso solicitado, finalizando el caso de uso.
Precondiciones:	
Flujo Normal de Eventos	
Acción del Actor	Respuesta del Negocio
<p>1- El caso de uso se inicia cuando el usuario llega al CASIE y solicita consultar un recurso.</p>	<p>2- El técnico general recibe la solicitud del usuario.</p> <p>3- El técnico general informa localización del recurso al usuario.</p>
<p>4- El usuario recibe la información.</p>	<p>5-El técnico general registra los datos de la solicitud en el registro de consulta de libro.</p>
Poscondiciones	Al usuario se le entregó un recurso y quedó registrado en el Registro Consulta de Libro.
Prioridad	Crítico

Tabla 7.Descripción textual del CUN_Solicitar recurso

Caso de Uso:	Localizar recurso.
Actores:	Usuario
Trabajadores:	Técnico general(inicia)
Resumen:	El caso de uso se inicia dada la solicitud de un recurso por un usuario y el técnico general empieza la gestión de localizar el recurso en los CASIE.

Capítulo 3: Presentación de la solución propuesta

Precondiciones:	Solicitar un recurso o solicitar seguir ruta de aprendizaje.
Flujo Normal de Eventos	
Acción del Actor	Respuesta del Negocio
	<ol style="list-style-type: none"> 1- El técnico general busca en el listado de recursos del CASIE. 2- Si el recurso aparece en el listado de recursos del CASIE el técnico general verifica la localización del recurso sino ver flujo alternativo 1. 3- Verifica el tipo del recurso encontrado. 4- Si el recurso no es digital informa al usuario la localización del recurso, sino ver flujo alternativo 2.
5- El usuario recibe la información, terminando así el caso de uso.	
Flujos Alternos 1	
Acción del Actor	Respuesta del Negocio
	<ol style="list-style-type: none"> 1- Si el recurso no aparece en el listado de recursos del CASIE se busca en los listados de recursos de los demás CASIE. 2- Si el recurso solicitado no se encuentra en ningún listado el técnico general informa al usuario que no disponen del recurso solicitado, sino ver flujo alternativo 3.
3- El usuario recibe la información.	
Flujos Alternos 2	
Acción del Actor	Respuesta del Negocio
	<ol style="list-style-type: none"> 1- Si el recurso es digital el técnico general verifica si el recurso está instalado. 2- Si el recurso está instalado informa al usuario la localización del recurso, sino ver flujo alternativo 5.
3- El usuario recibe la información, terminando así el caso de uso.	
Flujos Alternos 3	
Acción del Actor	Respuesta del Negocio
	1- Si el recurso se encuentra en el listado de otro CASIE, ir a la acción 2 del flujo normal de eventos.
Flujos Alternos 4	
Acción del Actor	Respuesta del Negocio
	<ol style="list-style-type: none"> 1- Si el recurso no está instalado comunica al técnico informático instalar el recurso. 2- Informa al usuario la situación.

3- EL usuario recibe la información, finalizando así el caso de uso.	
Poscondiciones	Al usuario se le localiza el recurso solicitado en el propio CASIE o en los demás CASIE de la UCI.
Prioridad	Crítico

Tabla 8. Descripción textual del CUN_Localizar recurso

Caso de Uso:	Instalar recurso.
Actores:	Usuario
Trabajadores:	Técnico Informático(Inicia), Técnico general
Resumen:	El caso de uso se inicia cuando el recurso solicitado por un usuario es digital y no está instalado y el técnico general orienta al técnico informático instalar el recurso.
Precondiciones:	
Flujo Normal de Eventos	
Acción del Actor	Respuesta del Negocio
	<ol style="list-style-type: none"> 1. El técnico informático recibe la orientación del técnico general. 2. El técnico informático comprueba poder instalar el recurso. 3. Si el recurso se puede instalar el técnico informático instala el recurso, sino ver flujo alterno 1. 4. El técnico informático informa la disponibilidad del recurso al técnico general. 5. El técnico general recibe confirmación de la instalación del recurso. 6. El técnico general informa al usuario que ya el recurso está instalado.
7. El usuario recibe la información de la disponibilidad del recurso solicitado, finalizando así el caso de uso.	
Flujos Alterno 1	
Acción del Actor	Respuesta del Negocio
	<ol style="list-style-type: none"> 1. Si el recurso solicitado no se puede instalar, el técnico informático informa al técnico general que el recurso solicitado no está disponible. 2. El técnico general recibe la información. 3. El técnico general informa al usuario.
4. El usuario recibe la información, finalizando así el caso de uso.	

Pos condiciones	Queda instalado el recurso.
Prioridad	Crítico

Tabla 9. Descripción textual del CUN_Instalar recurso

3.2.4 Diagramas de actividades

El diagrama de actividad es un grafo que contiene estados en que puede hallarse una actividad. Un estado de actividad representa la ejecución de una sentencia de un procedimiento, o el funcionamiento de una actividad en un flujo de trabajo. En vez de esperar un evento, como en un estado de espera normal, un estado de actividad espera la terminación de su cómputo. Cuando la actividad termina, entonces la ejecución procede al siguiente estado de actividad dentro del grafo. Una transición de terminación es activada en un diagrama de actividades cuando se completa la actividad precedente. Ver anexos.(19)

3.2.5 Modelo de Objetos

El diagrama de clases del Modelo de Objeto, es un artefacto que se construye para describir el modelo de objetos del negocio, muestra la participación de los trabajadores y entidades del negocio y la relación entre ellos. Ver anexos.

3.3 Especificación de los requisitos del software

La captura de requisitos es una parte esencial en el proceso de desarrollo de un software ya que evita cambios posteriores en el sistema y facilita el entendimiento con el cliente.

3.3.1 Requisitos Funcionales del Sistema

Visualizar datos de inscripción

1: Permitir visualizar datos de inscripción del usuario.

Mostrar usuario

R4: Mostrar expediente de un usuario dado un identificador.

R4.1: Mostrar datos generales del usuario.

R4.2: Mostrar registro de un usuario.

Registrar comportamiento indebido

R18: Detectar comportamientos indebidos del usuario.

R19: Registrar en el registro del usuario el comportamiento indebido.

R20: Ordenar los comportamientos indebidos en orden descendiente dada la cantidad de veces en las que se incurre en el comportamiento.

Gestionar usuario

R26.1: Crear un usuario.

R26.2: Modificar datos de inscripción de un usuario.

R26.3: Eliminar un usuario del sistema.

Verificar usuario

R30: Verificar datos de autenticación.

Ordenar los recursos

R5: Ordenar los recursos utilizados por el usuario en orden descendiente.

Generar reporte del usuario

R34: Generar reporte de los datos de todos los usuarios según el criterio de búsqueda especificado: nombre, apellidos, edad, sexo, provincia, municipio, aérea, nivel, estilo, asesor.

Crear listado de usuarios pendientes

R33: Crear listado de usuarios pendientes de aceptación de asesoría.

Verificar existencia de ruta concluida

R72: Verificar existencia de ruta de aprendizaje concluida dado un usuario.

Asignar rol

R92 Asignar rol.

R92.1 Asignar permisos al rol

Mostrar listado de usuarios

R37: Mostrar listado al administrador de usuarios con sus asesores correspondientes dado un rango de fechas.

Actualizar registro del usuario

R35: Actualizar registro del usuario.

Enviar correo

R8: Enviar por correo a un usuario con 24 horas de antelación las próximas actividades según de la ruta.

R6: Permitir enviar correos a un usuario.

Crear listado de actividades

R85: Crear listado de actividades no cumplidas de una ruta de aprendizaje dado un usuario.

Construir perfil de usuario

R67: Construir perfil de usuario.

Asignar permisos

R69: Permitir al usuario seleccionar un servicio.

R71: Permitir al usuario seleccionar un recurso didáctico.

R81: Permitir al usuario seleccionar una evaluación.

R83: Permitir al usuario solicitar un recorrido.

R86: Permitir al usuario seleccionar una actividad.

3.3.2 Requerimientos no Funcionales del Sistema

Los requerimientos no funcionales son propiedades o cualidades que el producto debe tener, son fundamentales para el buen funcionamiento del sistema, aquí es donde radica su importancia. A continuación se les muestra los requerimientos de la aplicación, los cuales están escritos por clasificación.

Usabilidad.

RNF 1: Para utilizar la aplicación es necesario estar registrado en la aplicación según el rol disponible, para facilitar una mejor utilización de los datos y acceso de información.

RNF 2: El módulo de administración debe permitir la petición concurrente de un gran número de solicitudes de los otros módulos.

Portabilidad.

RNF 3: La aplicación podrá ser usada bajo el sistema operativo Windows, Linux.

Apariencia o interfaz externa.

RNF 4: La aplicación que se utilizará debe ser lo más interactiva posible, brindará una interfaz con poca complejidad para que el usuario no tenga dificultad al utilizarlo.

Software.

RNF 5: Para la implantación del sistema se requiere de un servidor perteneciente a la familia NT.

RNF 6: Para la utilización del sistema por lado del cliente se hace necesario tener disponible un navegador web.

RNF 7: Se utiliza Apache como servidor web V1.X o superior y MySQL server V5.0.22 como servidor de bases de Datos.

Hardware.

RNF 8: Las computadoras clientes requieren como mínimo un procesador Pentium II, con 64 Mb de memoria RAM, mínimo, recomendable. Deben estar conectadas en red con el servidor.

RNF 9: Para los servidores (Web, SGBD) se hace necesario: PENTIUM II, con 512 MB de RAM y para el almacenamiento de la información 40Gb de capacidad de disco, mínimo, recomendable.

Rendimiento.

RNF 10: El sistema debe ser lo más eficiente posible para poder lograr un tiempo de respuesta adecuado.

Soporte.

RNF 11: Diccionarios digitales o enlaces a sitios de diccionarios.

RNF 12: Garantizar la prueba del sistema, además de dar un adiestramiento a las personas que tendrán la responsabilidad de administrar el sistema.

Seguridad.

RNF 13: Mantener la integridad de la información.

RNF 14: Permitir el almacenamiento de la información por un período de tiempo de 30 días.

RNF 15: Permitir que las contraseñas de los usuarios se almacenen y viajen por la red encriptados.

RNF 16: Garantizar que la contraseña sea fuerte, que cumpla con las exigencias de longitud y composición.

RNF 17: Garantizar que la contraseña sea cambiada cada sesenta días.

RNF 18: Advertir al usuario sobre acciones “irreversibles”.

RNF 19: Asignar el rol de usuario, por defecto al crear un usuario.

3.4 Descripción del Sistema Propuesto

El sistema CEVALE está dividido en seis módulos cada módulo cuenta con un diagrama de caso de uso del sistema, a continuación se representa el correspondiente al módulo de administración.

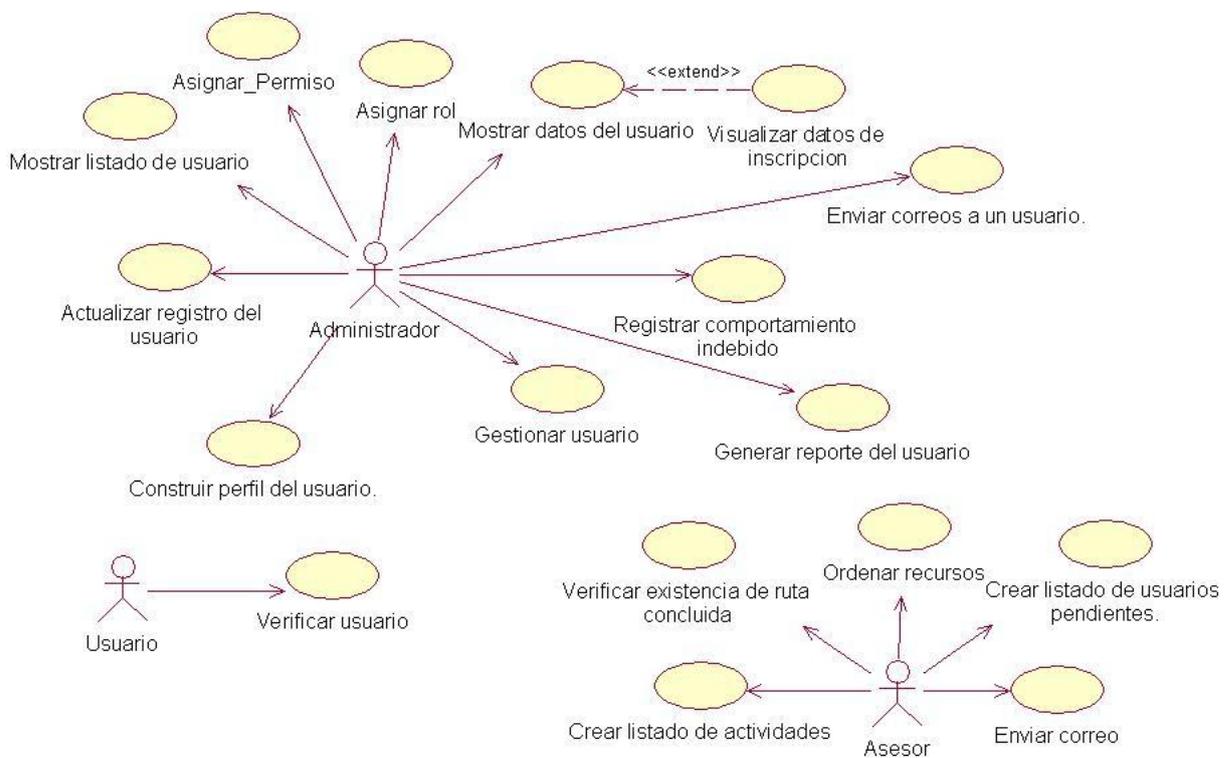


Fig. 2. Diagrama de CU de sistema del Módulo de Administración

3.4.1 Descripción de los actores

Actores del sistema	Justificación
Administrador	Es el encargado de controlar toda la información referente a los usuarios y de administrar el sistema.
Asesor	Es el responsable de controlar que se cumplan las actividades de la ruta de aprendizaje.
Usuario	Es el encargado de presentar sus credenciales para acceder al sistema.

Tabla 10. Actores del Sistema

3.4.2 Casos de Uso del Sistema

Debido a la gran cantidad de casos de uso que presenta el sistema y para lograr una mejor organización del trabajo se decidió mostrar en este epígrafe solo los casos de uso críticos.

Caso de Uso:	Gestionar usuarios.
Actores:	Administrador (inicia).
Propósito	Permite al administrador gestionar (crear, modificar, eliminar) los usuarios.
Resumen:	El caso de uso se inicia cuando el Administrador selecciona la opción de Gestionar Usuario, luego selecciona el tipo de gestión, introduce los datos necesarios, el sistema realiza la acción seleccionada por el Administrador y termina el CUS.
Referencia	R26, R26.1, R26.2, R26.3.
Precondiciones:	El administrador debe estar autenticado en el sistema.
Pos condiciones	Nuevo usuario registrado en el sistema. Datos del usuario modificado en el sistema. Datos del usuario eliminado del sistema.
CU relacionados	
Flujo Normal de Eventos	
Acción del Actor	Respuesta del Sistema
1. El caso de uso comienza cuando el Administrador selecciona la opción de Gestionar Usuario.	2. El sistema le muestra un formulario con las opciones de Crear Usuario, Modificar Usuario y Eliminar Usuario.
3. El Administrador selecciona una opción:	

<p>a. Si selecciona Crear Usuario, véase sección <i>Crear Usuario</i>.</p> <p>b. Si selecciona Modificar, véase sección <i>Modificar datos de inscripción de un usuario</i>.</p> <p>c. Si selecciona Eliminar, véase sección <i>Eliminar un usuario</i>.</p>	
Sección “Crear Usuario”	
1. El Administrador selecciona la opción crear un usuario.	2. El sistema muestra el formulario para crear un usuario.
3. El Administrador introduce los datos del usuario.	<p>4. El sistema verifica que los datos del usuario no exista en la base de datos.</p> <p>5. Si el usuario especificado existe véase flujo alternativo 5.1.</p> <p>6. Si el usuario especificado no existe, el sistema crea el usuario.</p> <p>7. El sistema muestra un mensaje que confirma la exitosa ejecución de dicha acción.</p>
8. El Administrador recibe la notificación y termina así el CUS.	
Flujos Alternos	
	5.1 Si se verifica que el usuario especificado existe en la base de datos se muestra un mensaje de error.
Sección “Modificar datos de inscripción de un usuario”	
1. El Administrador selecciona la opción modificar un usuario.	2. El sistema le muestra un listado de todos los usuarios.
3. El Administrador selecciona el usuario que desea modificar sino ir al flujo alternativo 1 de la sección modificar.	4. El sistema muestra los datos actuales del usuario seleccionado que pueden ser modificados.
5. El Administrador modifica alguno o todos los datos.	<p>6. El sistema verifica si los datos fueron introducidos correctamente.</p> <p>7. Si los datos fueron introducidos incorrectamente, véase flujo alternativo 7.1.</p> <p>8. Si los datos fueron introducidos correctamente, el sistema modifica los datos del usuario.</p> <p>9. El sistema muestra un mensaje que confirma la exitosa ejecución de dicha acción.</p>

10. El Administrador recibe la notificación y termina así el CUS.	
Flujos Alternos Sección Modificar.	
Acción del Actor	Respuesta del Sistema
	1. Si el Administrador no seleccionó el usuario que desea modificar se muestra un mensaje de error.
	7.1. El sistema muestra un mensaje de error.
Sección “Eliminar un usuario.”	
Acción del Actor	Respuesta del Sistema
1. El Administrador selecciona la opción eliminar un usuario.	2. El sistema le muestra un listado de todos los usuarios.
3. El Administrador selecciona el usuario que desea eliminar sino ir al flujo alternativo 1.	4. El sistema muestra un mensaje de advertencia, solicitando que se confirme si se desea realmente eliminar el usuario seleccionado.
5. El Administrador confirma que desea eliminar el usuario seleccionado.	6. El sistema elimina la información del usuario seleccionado.
Flujos Alternos Sección Eliminar	
Acción del Actor	Respuesta del Sistema
	1. Si el Administrador no seleccionó el usuario que desea eliminar se muestra un mensaje de error.
Prioridad	Crítico

Tabla 11. Descripción del CUS_Gestionar Usuario

Caso de Uso:	Verificar usuario.
Actores:	Usuario (inicia).
Propósito	Reconocer la identidad del usuario al sistema.
Resumen:	El usuario se identifica ante el sistema el cual valida sus datos.
Referencia	R30.
Precondiciones:	Deben existir usuarios en el sistema.
Pos condiciones	El usuario puede trabajar con el sistema sin volver a identificarse para cada operación que realice.
CU relacionados	
Flujo Normal de Eventos	
Acción del Actor	Respuesta del Sistema
1. El usuario introduce su nombre y su contraseña.	2. El sistema verifica en la base de datos los datos introducidos por el usuario .Si los datos son correctos el sistema permite entrar al mismo.
Flujos Alternos	
Acción del Actor	Respuesta del Sistema
	1. Si se introdujeron los datos incorrectos el sistema envía un mensaje de error.
Prioridad	Crítico

Tabla 12.Descripción del CUS_Verificar Usuario

Caso de uso:	Asignar rol.	
Actores:	Administrador (inicia)	
Propósito:	Asignar un rol específico a un usuario del sistema que cumplirá esta función.	
Resumen:	El Administrador asigna posible roles a un usuario determinado, selecciona dicho usuario, el sistema facilita entonces el listado de todos los roles que se han especificado para el sistema, de donde el Administrador seleccionará el indicado para asignarle finalmente a ese usuario.	
Referencias:	R92, R92.1	
Precondiciones:	El Administrador se encuentra previamente autenticado en el sistema.	
Pos condiciones:		
CU relacionados:		
Flujo Normal de Eventos		
Acción del actor:	Respuesta del sistema:	
1. El Administrador selecciona la opción asignar rol.	2. El sistema muestra el listado con todos los usuarios.	
3. El Administrador selecciona el usuario al que desea asignarle un rol, sino ir al flujo alterno 3.1.	4. El sistema muestra un formulario para que se seleccione el módulo al que pertenece el rol que se desea asignar.	
5. El Administrador selecciona el módulo indicado, sino ir al flujo alterno 5.1.	6. El sistema muestra el listado de todos los roles del módulo que pueden ser asignados a dicho usuario.	
7. El Administrador selecciona uno o más roles que desea asignar al usuario, sino ir al flujo alterno 6.1.	8. El sistema asigna el rol seleccionado al usuario y muestra un mensaje que confirma el desarrollo exitoso de la operación especificada.	
9. El Administrador recibe la notificación y así termina el CUS.		
Flujos Alternos		
Acción del actor:	Respuesta del sistema:	
	3.1 Si el Administrador no seleccionó un usuario al que desea asignarle un nuevo rol, el sistema muestra un mensaje de error.	
	5.1. Si el Administrador no seleccionó un módulo al que pertenece el rol que se desea asignar, el sistema muestra un mensaje de error.	
	6.1. Si el Administrador no seleccionó uno o más roles que desea asignar al usuario, el sistema muestra un mensaje de error.	
Prioridad	Crítico.	

Tabla 13. Descripción del CUS_Asignar Rol

Caso de Uso:	Construir perfil de usuario.	
Actores:	Administrador (inicia)	
Propósito	Mediante este caso de uso el sistema permite crear nuevos perfiles de usuarios.	
Resumen:	El caso de uso se inicia cuando el administrador introduce los datos del usuario para crear un perfil y termina una vez creado este perfil.	
Referencia	R67	
Precondiciones:	No exista un usuario con el mismo perfil en el sistema.	
Pos condiciones	Se registra el perfil del nuevo usuario.	
CU relacionados		
Flujo Normal de Eventos		
Acción del Actor	Respuesta del Sistema	
1- El administrador del sistema selecciona la opción de construir un perfil.	2- El sistema le muestra al administrador del sistema un formulario a completar donde le pide los datos para construir el perfil del usuario.	
3- El administrador del sistema introduce los datos solicitados por el sistema.	4- El sistema verifica los datos entrados por el administrador del sistema.	
	5- Si los datos introducidos son correctos, el sistema comprueba si el usuario ya existe en la Base de Datos, sino ir a flujo alterno 5.1.	
	6- Si el usuario no existe, el sistema agrega el usuario con su perfil en la Base de Datos y termina el CUS, sino ir al flujo alterno 6.1.	
Flujos Alternos		
Acción del Actor	Respuesta del Sistema	
	5.1 Si los datos introducidos son incorrectos, el sistema muestra un mensaje de error indicando donde está el dato incorrecto e indica al administrador del sistema retornar a la acción 3.	
	6.1- Si el usuario existe en el sistema, el sistema le muestra al administrador del sistema un mensaje de que el usuario ya existe en la Base de Datos y termina el CUS.	
Prioridad	Crítico.	

Tabla 14. Descripción del CUS_Construir Perfil del Usuario

Caso de Uso:	Asignar Permisos
Actores:	Administrador (inicia)
Propósito	Asignar un permiso a un rol específico del sistema que cumplirá con los privilegios otorgados.
Resumen:	El Administrador asigna un permiso a un rol determinado, el sistema facilita entonces el listado de todos los roles que se han especificado para el sistema, de donde el Administrador seleccionará el indicado para asignarle finalmente el permiso a ese usuario.

Referencia	R69, R71,R81,R83,R86	
Precondiciones:	El administrador debe estar autenticado en el sistema.	
Pos condiciones		
CU relacionados		
Flujo Normal de Eventos		
Acción del Actor	Respuesta del Sistema	
1. El caso de uso comienza cuando el Administrador desea asignarle un permiso a un rol del sistema.	2. El sistema le muestra un listado con todos los roles existentes en la base de datos.	
3. El Administrador selecciona el rol deseado, sino ir al flujo alternativo 3.1.	4. El sistema le muestra un listado con todos los permisos: a. "Permitir al usuario seleccionar un recurso didáctico". b. "Permitir al usuario seleccionar una evaluación". c. "Permitir al usuario seleccionar un recorrido". d. "Permitir al usuario seleccionar un servicio". e. "Permitir al usuario seleccionar una actividad".	
5. El Administrador selecciona uno o más permisos, sino ir al flujo alternativo 5.1	6. El sistema asigna el permiso seleccionado y muestra un mensaje de ejecución satisfactoria.	
7. El Administrador recibe la notificación y termina así el CUS.		
Flujo Alternos		
Acción del actor	Acción del sistema	
	3.1. Si el Administrador no seleccionó el rol, el sistema muestra un mensaje de error.	
	5.1. Si el Administrador no seleccionó el permiso, el sistema muestra un mensaje de error.	
Prioridad	Crítico.	

Tabla 15. Descripción del CUS_Asignar Permisos

3.5 Conclusiones

En este capítulo se modeló el proceso de funcionamiento del sistema a través del Modelo del Negocio, el cual permitió comprender los procesos básicos del funcionamiento de los CASIE. Queda evidenciado que la captura de los requerimientos es un paso esencial para el análisis del sistema, dando así una visión del futuro de la aplicación. La identificación de los casos de uso críticos permite dar prioridad en la realización de estos.

CAPÍTULO 4: “Construcción de la solución propuesta”

4.1 Introducción

En este capítulo se abordará uno de los flujos de trabajo de RUP, el análisis y diseño. Aunque esta metodología une al análisis y diseño en un solo flujo de trabajo ambos tienen funcionalidades diferentes. Estos dos procesos son muy importantes en el desarrollo de software porque constituyen la vista lógica de la arquitectura del software, la cual es un epígrafe más de este capítulo. Para finalizar se realizan los diagramas de componentes de la aplicación y se desarrolla el Modelo de Despliegue.

4.2. Arquitectura a implementar

La Arquitectura de Software es un tema de la actualidad muy importante que ha dominado prácticamente toda la década del 90, fue tomando valor significativo a medida que los programadores se fueron dando cuenta de la importancia que representaba para la creación de un software, porque da una visión general del proyecto que se está desarrollando y permite comprender bien el sistema.

"La Arquitectura de Software es la organización fundamental de un sistema encarnada en sus componentes, las relaciones entre ellos y el ambiente y los principios que orientan su diseño y evolución."

4.3 Arquitectura en capas

La arquitectura por capas tiene como objetivo primordial la separación de la lógica del negocio con la lógica de diseño, un ejemplo básico de esto es separar la capa de datos de la capa de presentación al usuario.

La ventaja principal, es que el desarrollo se puede llevar a cabo en varios niveles y en caso de algún cambio sólo se ataca al nivel requerido sin tener que revisar entre código mezclado.

Además permite distribuir el trabajo de creación de una aplicación por niveles, de este modo, cada grupo de trabajo está totalmente abstraído del resto de los niveles.

4.3.1 Arquitectura en tres capas

1. *Capa de presentación:* Es la capa que contiene todos los aspectos del sistema, por ejemplo las interfaces, además es la que interactúa con el usuario, le comunica la información y captura la información del usuario 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.

2. *Capa de lógica del negocio:* Es donde residen los programas que se ejecutan, recibiendo las peticiones del usuario y enviando las respuestas tras el proceso. Se denomina capa de negocio (e incluso de lógica del negocio) pues es aquí donde se establecen todas las reglas que deben cumplirse. Esta capa brinda servicios a la capa de presentación, y se comunica con la capa de datos, para solicitar, almacenar o actualizar los datos.

3. *Capa de acceso a datos:* Es donde se almacena toda la información que se encuentra en el sistema, además recibe las solicitudes de almacenamiento o recuperación de información desde la capa de negocio.

Ventajas:

- Soporta un diseño basado en niveles de abstracción crecientes, lo cual a su vez permite a los implementadores la partición de un problema complejo en una secuencia de pasos incrementales.
- Admite muy naturalmente optimizaciones y refinamientos.
- Proporciona amplia reutilización. Al igual que los tipos de datos abstractos, se pueden utilizar diferentes implementaciones o versiones de una misma capa en la medida que soporten las mismas interfaces de cara a las capas adyacentes. Esto conduce a la posibilidad de definir interfaces de capa estándar, a partir de las cuales se pueden construir extensiones o prestaciones específicas.

Desventajas:

- No admite un buen mapeo en una estructura jerárquica. Incluso cuando un sistema se puede establecer lógicamente en capas, consideraciones de rendimiento pueden requerir acoplamientos específicos entre capas de alto y bajo nivel.

- A veces es también extremadamente difícil encontrar el nivel de abstracción correcto.
- Los cambios en las capas de bajo nivel tienden a filtrarse hacia las de alto nivel, en especial si se utiliza una modalidad relajada; también se admite que la arquitectura en capas ayuda a controlar y encapsular aplicaciones complejas, pero complica no siempre razonablemente las aplicaciones simples.

Se decidió utilizar la arquitectura en tres capas donde cada capa es nombrada: Capa de acceso a datos, Capa de lógica del negocio y Capa de presentación respectivamente, aunque es posible continuar subdividiendo este modelo en subcapas para una mayor flexibilidad en la distribución en el equipo de desarrollo y durante el mantenimiento.

4.4 Patrones de diseño

“Los patrones conducen a arquitecturas más pequeñas, más simples y más comprensibles.”

G. Booch

Un patrón es una pareja de problema/solución con un nombre que estandariza los principios y las sugerencias. Es aplicable a otros contextos, con una sugerencia sobre la manera de usarlo en diferentes situaciones. El objetivo de los patrones es crear un lenguaje común a una comunidad de desarrolladores para comunicar experiencia sobre los problemas y sus soluciones.

Se conocen fundamentalmente cinco patrones de diseño que son utilizados para la asignación de responsabilidades llamados Patrones GRASP. Estos son:

Experto: Es un principio básico que suele utilizarse en el diseño. Expresa que los objetos realizan operaciones relacionadas con la información que poseen. Con el uso de este patrón se conserva el encapsulamiento, ya que los objetos se valen de su propia información, para hacer lo que se les pide.

Creador: Guía la asignación de responsabilidades relacionadas con la creación de objetos. El propósito fundamental de este patrón es encontrar un creador para conectarlo con el objeto producido en cualquier evento.

Alta Cohesión: Cada elemento del diseño debe realizar una labor única dentro del sistema, no desempeñada por el resto de los elementos y auto-identificables. Un ejemplo de ello es cuando se crean los paquetes de servicios o clases agrupadas por funcionalidades que pueden ser fácilmente reutilizables bien por uso directo o por herencia.

Bajo Acoplamiento: Es un principio que se debe recordar durante las decisiones de diseño. Es un patrón evaluativo que el diseñador debe aplicar al juzgar sus decisiones de diseño, además, soporta el diseño de clases más independientes que reducen el impacto de los cambios, y más reutilizables, que acrecientan la oportunidad de una mayor productividad.

Por último se tiene el patrón controlador, el cual es el que guiará fundamentalmente el diseño de los casos de uso del sistema.

El *patrón controlador* durante este flujo de trabajo se le asigna la responsabilidad de las operaciones del sistema. La misma clase controlador debería usarse con todos los eventos sistémicos de un caso de uso, para así conservar la información referente al estado del caso de uso. Esto es útil por ejemplo para detectar eventos del sistema fuera de secuencia. Un controlador es un objeto de interfaz que se encarga de manejar un evento del sistema. Define además el método de su operación. (20)

Un defecto frecuente al diseñar controladores consiste en asignarles demasiada responsabilidad. Normalmente un controlador debería delegar a otros objetos el trabajo que ha de realizarse mientras coordina la actividad.

La primera categoría de controlador, es un controlador de fachada, que representa al sistema global. Es una clase que, para el diseñador, representa de alguna manera al sistema entero. Si se recurre a la cuarta categoría de controlador un manejador artificial de casos de uso, habrá entonces un controlador para cada caso. Este no es un objeto del dominio, es un concepto artificial.

Un controlador de casos de uso es una buena alternativa cuando hay muchos eventos de sistema entre varios procesos: asigna su manejo a clases individuales controlables.(20)

Este patrón tiene un mayor potencial de los componentes reutilizables ya que garantiza que los procesos de dominio sean manejados por la capa de los objetos del dominio y no por la de interfaz. Desde el punto de vista técnico, las responsabilidades del controlador podrían cumplirse en un objeto de interfaz, pero esto supone que el código del programa y la lógica relacionada con la realización de

los procesos del dominio puro quedarían incrustados en los objetos interfaz. Un diseño de interfaz como controlador reduce la posibilidad de reutilizar la lógica de los procesos del dominio en aplicaciones futuras, por estar ligada a una interfaz determinada. En cambio, el hecho de delegar a un controlador la responsabilidad de la operación de un sistema entre las clases del dominio soporta la reutilización de la lógica para manejar los procesos a fines del negocio en aplicaciones futuras.

4.5 Análisis

En el análisis podemos estructurar los requisitos de manera que nos facilite su comprensión, su preparación, su modificación y en general su mantenimiento. Esta estructura (basada en clases de análisis y paquetes) es independiente de la estructura que se dio a los requisitos (basada en casos de uso). Sin embargo existe una trazabilidad directa entre esas distintas estructuras, la cual se define entre casos de uso del modelo de casos de uso y realizaciones del caso de uso en el modelo de análisis. Las actividades de análisis son desarrolladas con el objetivo de facilitar la entrada al diseño, por lo que son un paso inicial y una primera aproximación conceptual para una vez comprendido los requisitos a este nivel, aumentar el nivel de especificidad en aras de garantizar el cubrimiento de los requisitos funcionales y no funcionales, considerando además el entorno de implementación. (21)

4.5.1 Diagrama de Clases del Análisis

Una de las principales salidas o artefactos del análisis es el Diagrama de Clases del Análisis. En este se representan las clases de análisis (clase Interfaz, clase controlador y clases entidad), sus relaciones entre sí y con los actores del sistema.

En una aplicación donde la arquitectura es en tres capas como la que se está desarrollando, en la capa de presentación se encuentran las clases interfaz porque es donde se ejecutan las aplicaciones del cliente, en la capa de lógica del negocio las clases de control porque es donde ocurren todos los procesos y en la capa de acceso a datos se encuentran las clases entidad ya que allí es donde se accede a la base de datos.

Por la complejidad de los diagramas de clases de la aplicación, no se debe realizar un solo diagrama de clases del análisis para todo el sistema, ya que resulta complejo de comprender. Ver anexos.

4.6 Diseño

El diseño es el flujo de trabajo que permite la comprensión de los aspectos relacionados con los requisitos no funcionales y restricciones relacionadas con los lenguajes de programación, componentes reutilizables, sistemas operativos, y tecnologías de interfaz de usuario y es el que da la entrada a las actividades de implementación, capturando los requisitos o subsistemas individuales, interfaces y clases.(21)

4.6.1 Diagrama de Clases del Diseño

En el diseño se modela el sistema teniendo en cuenta la arquitectura para que soporte todos los requisitos funcionales, no funcionales y las restricciones que se le suponen. Una entrada esencial en el diseño es el resultado del análisis, o sea, el modelo de análisis, que proporciona una comprensión detallada de los requisitos. Además impone una estructura del sistema, la cual hay que conservar lo más posible cuando se le da forma a la aplicación.(21)

Concretamente se puede definir como propósitos del diseño:

- ✚ Transformar los requerimientos en un diseño de cómo el sistema debe ser.
- ✚ Desarrollar una robusta arquitectura del sistema.
- ✚ Adaptar el diseño para que se corresponda con el entorno de implementación, diseñando sus funcionalidades.

El diagrama de clases del diseño describe la realización de los casos de uso, y sirve como una abstracción del modelo de implementación y el código fuente. Es usado como una entrada inicial en las actividades de implementación y prueba. Ver anexos.

4.7 Modelo de Despliegue

El modelo de despliegue muestra la configuración de los nodos de procesamiento en tiempo de ejecución, los vínculos de comunicación entre ellos, y las instancias de los componentes y objetos que residen en ellos. Consiste en:

Nodos: Elementos de procesamiento con al menos un procesador, memoria, y posiblemente otros dispositivos.

Dispositivos: Nodos estereotipados sin capacidad de procesamiento en el nivel de abstracción que se modela.

Conectores: Expresa el tipo de conector o protocolo utilizado entre el resto de los elementos del modelo.

El propósito del modelo de despliegue es capturar la configuración de los elementos de procesamiento, y las conexiones entre estos elementos en el sistema. El modelo de despliegue también mapea procesos dentro de estos elementos de procesamiento, permitiendo la distribución del comportamiento a través de los nodos que son representados.

Se utiliza para capturar los elementos de configuración del procesamiento y las conexiones entre esos elementos. También se utiliza para visualizar la distribución de los componentes de software en los nodos físicos. Ver anexos.

4.8 Conclusiones

Con el desarrollo de este capítulo, se logró definir el análisis y el diseño de la aplicación, obteniendo así, los diagramas de clases de cada flujo donde se representaron las clases y sus asociaciones. Se plantean propósitos del diseño que ayudan a mejorar la futura implementación del sistema. Se realizó el diagrama de despliegue, dando una explicación de en qué consiste dicho diagrama. Todos estos elementos obtenidos brindan una idea más clara de la estructura del sistema e influyen en la implementación de una mejor herramienta. También se trata de la arquitectura y el patrón de diseño utilizado para llevar a cabo el diseño de la aplicación.

Conclusiones Generales

El proyecto CEVALE presenta un importante valor social pues está encaminado a mejorar las condiciones de trabajo de los especialistas que laboran en el Departamento Central de Idiomas Extranjeros.

A través del estudio que se llevó a cabo para lograr entender el proceso de negocio, se detectaron las deficiencias que presentan los CASIE, y en general, el Departamento Central de Idiomas Extranjeros, lo que constituyó el punto de partida de esta investigación, para así poder diseñar el módulo de administración.

Uno de los principales aspectos a tener en cuenta fue la demostración de la eficacia de los lenguajes y tecnologías propuestas para el desarrollo del sistema, expresando los resultados en términos de los diferentes modelos de la ingeniería de software que se generan de la aplicación de la metodología seleccionada. Se llega a la determinación de que los requerimientos y casos de uso identificados satisfacen las necesidades funcionales exigidas. Además, se logrará la seguridad y protección de los datos consecuente con el nivel requerido.

El sistema estará basado en una arquitectura de tres capas, utilizando el patrón de diseño controlador, respetando siempre, los principios básicos de diseño descritos para el desarrollo del sistema.

En resumen, se logró diseñar completamente el Módulo de Administración del sistema CEVALE, por lo que se concluye que los objetivos propuestos del presente trabajo han sido cumplidos satisfactoriamente.

Recomendaciones

Se recomienda incorporar para futuras versiones, comunicación con la base de datos Ciudadanos a través de sus servicios web, para consultar información o datos de los usuarios de la comunidad UCI.

Se recomienda que se incorporen requerimientos para la gestión de los roles y los permisos del sistema.

Se sugiere que se incorpore la funcionalidad de deshabilitar usuarios cuando sea necesario.

Referencias Bibliográficas y Webgrafía

1. AUTORES, C. D. "Introducción a la Seguridad Informática". En Seguridad Informática. Conferencia 1. Universidad Ciencias Informáticas. 2008.
- 2.------. "Control de acceso. Identificación y autenticación". En Seguridad Informática. Conferencia 2. Universidad Ciencias Informáticas. 2008.
3. GONZALO ÁLVAREZ MARAÑÓN, C. "Autenticación y Autorización, Control de acceso por certificados". 2002, Disponible en: <http://www.iec.csic.es/criptonomicon/autenticacion/cert.html>.
- 4.GONZALO ÁLVAREZ MARAÑÓN, C. "Ventajas e inconvenientes del control de acceso por certificados" 2002, Disponible en: <http://www.iec.csic.es/criptonomicon/autenticacion/certvei.html>.
- 5.------. "Autenticación y Autorización, Ventajas e inconvenientes del control de acceso por nombre de usuario y contraseña". 2002, Disponible en: <http://www.iec.csic.es/criptonomicon/autenticacion/autvei.html>.
6. AUTORES, C. D. "Protección contra programas malignos". En Seguridad Informática. Conferencia 4. Universidad Ciencias Informáticas. 2008.
- 7.MANIVESA, C. "Inyección SQL". 2004, Disponible en: <http://sql.manivesa.com/Tutoriales+SQL/Programacion+.NET/150.aspx>.
8. AUTORES, C. D. "Ataques informáticos y mecanismos de defensa". En Conferencia 3. Cuba. 2006-2007.
- 9.SPIDYNAMICS. "SPIDynamics, now part of HP Software", AMP. 2007, Disponible en: <http://www.spydynamics.com/products/amp/datasheet.html>
- 10.------. "SPIDynamics, now part of HP Software".DevInspect. 2007, Disponible en: <http://www.spydynamics.com/products/devinspect/index.html>.
- 11.------. "JA-SIG Announces Immediate Availability of uPortal 2.6 and Central Authentication Service (CAS) 3.1". 2004, Disponible en: http://www.ja-sig.org/news/uportal_260_release.html.

12. DESARROLLOWEB. "Características de PHP5". Disponible en: <http://www.desarrolloweb.com/articulos/1699.php>.
- 13.-----"¿Qué es Eclipse?" n° Disponible en: http://64.233.179.104/translate_c?hl=es&sl=en&u=http://www.eclipse.org/home/newcomers.php&prev=/search%3Fq%3Declipse%26hl%3Des.
14. JOSÉ H. CANÓS, P. L., M^a CARMEN PENADÉS CARMEN PENADÉS. "Metodologías Ágiles en el Desarrollo de Software". Disponible en: <http://www.willydev.net/descargas/prev/TodoAgil.Pdf>.
15. SANCHEZ, M. M. "Metodologías De Desarrollo De Software". 2004, Disponible en: http://www.informatizate.net/articulos/pdfs/metodologias_de_desarrollo_de_software_07062004.pdf.
16. JAMES RUMBAUGH, I. J., GRADY BOOCH. "El Lenguaje Unificado de Modelado. Manual de referencia". 528 p.
17. MYSQL. "Panorámica del sistema de gestión de base de datos MySQL". 2008, Disponible en: <http://dev.mysql.com/doc/refman/5.0/es/what-is.html>.
- 18.-----"Las principales características de MySQL". 2008 Disponible en: <http://dev.mysql.com/doc/refman/5.0/es/features.html>.
19. AUTORES, C. D. "Fase de Inicio. Modelo del Negocio". En Ingeniería Software I. Conferencia 2. Modelación del Negocio. Universidad Ciencias Informáticas. 2008.
20. LARMAN, C. "UML y Patrones. Introducción al análisis y diseño orientado a objetos". 507 p.
21. AUTORES, C. D. "Fase de Inicio. Flujo de Análisis y Diseño. Modelo de Análisis." En Ingeniería Software II. Conferencia 6 Universidad Ciencias Informáticas. 2008.

Bibliografía Consultada

1. CASTILLO, Á. D. "Webs dinámicas con PHP". 2008, Disponible en: <http://www.mailxmail.com/curso/informatica/phpprogramacionweb/capitulo4.htm>.
2. FÉLIX ÓSCAR GARCÍA RUBIO, C. B. S. "Metodologías de Desarrollo de Software". Disponible en: http://alarcos.inf-cr.uclm.es/per/fgarcia/isoftware/doc/tema3_1xh.pdf.
3. GUERRERO, L. A. "Análisis y Diseño Orientado a Objetos. Patrones para asignación de responsabilidades".
4. IVAR JACOBSON, G. B., JAMES RUMBAUGH. "El Proceso Unificado de Desarrollo de Software". 464 p. ISBN 84-7829-036-2.
5. JAMES RUMBAUGH, I. J., GRADY BOOCH. "El Lenguaje Unificado de Modelado. Manual de referencia". 528 p.
6. LARMAN, C. "UML y Patrones. Introducción al análisis y diseño orientado a objetos". 507 p.
7. S.PRESSMAN, R. "Ingeniería de Software: un enfoque práctico". Vol. 5 edición, 601 p.
8. SCHAMULLER, J. "Aprendiendo UML en 24 horas". 2000. 448 p. ISBN 968-444-463-X.

Glosario de Términos

A

Asesor del CASIE: Profesional académico que se encarga de emitir una calificación de las encuestas de nivel aplicada a los usuarios, así como orientarlos con las posibles rutas a seguir para su autoaprendizaje, utilizando los elementos didáctico-pedagógicos en la conducción del mismo.

Autoaprendizaje: Es el proceso al que se somete el usuario con el interés de aprender alguna cuestión teórica o técnica siguiendo una ruta de aprendizaje según su nivel y fase orientado o no por un asesor, con la conciencia de que deberá lograrlo poniendo su máximo empeño en ello y de que lo hará por sus propios medios, en tiempos que él decida.

C

Comportamiento indebido: Respuestas en el chat o foro dadas por un usuario que no esté dentro del contenido de la ruta de aprendizaje o algún tema relacionado con el aprendizaje de las lenguas extranjeras. Así como expresarse incorrectamente con el asesor o cualquier otro trabajador del CASIE.

Contrato: Es un documento que contiene los datos del usuario, su ruta de aprendizaje, el horario que debe cumplir para la realización de las actividades de la ruta de aprendizaje y el asesor que lo atiende.

CASIE: Los Centros de Autoaprendizaje y Servicios de Idiomas Extranjeros (CASIE) en la UCI ofrecen a estudiantes, profesores y a la comunidad universitaria en su conjunto una nueva experiencia en el autoaprendizaje de idiomas. Los CASIE son espacios físicos y virtuales apropiados para llevar a cabo la autogestión del aprendizaje. Su propósito es proporcionar al estudiante de idiomas todo lo que necesita para estudiar la lengua que desee: recursos didácticos, tecnologías y asesorías a través de entornos de aprendizaje altamente efectivos.

Cracker: Es un tipo de programa que realiza una modificación permanente o temporal sobre otro programa o en su código, para obviar una limitación o candado impuesto a propósito por el programador original.

E

Expediente: Documento que va a contener el registro y contrato del usuario. El asesor debe tener a mano el Expediente del usuario para determinar su ruta de aprendizaje.

F

Fase: El usuario que haya realizado las encuestas de estilos, necesidades y estrategias recibirá en que fase de aprendizaje se encuentra y por donde debe empezar su autoaprendizaje. Las fases han de corresponderse con los semestres impartidos según el actual plan de estudios. Serían 5 fases: Fase 1 (primer semestre), fase 2(segundo semestre), fase 3 (tercer semestre), fase 4 (IV semestre), fase 5 (V semestre).

Framework: Es una estructura de soporte definida en la cual otro proyecto de software puede ser organizado y desarrollado. Puede incluir soporte de programas, bibliotecas y un lenguaje interpretado entre otros software para ayudar a desarrollar y unir los diferentes componentes de un proyecto. Representa una arquitectura de software que modela las relaciones generales de las entidades del dominio. Provee una estructura y una metodología de trabajo la cual extiende o utiliza las aplicaciones del dominio.

N

Nivel: El usuario que haya realizado las encuestas de estilos, necesidades y estrategias recibirá un nivel determinado dependiendo del resultado obtenido, este puede ser: nivel básico, intermedio o avanzado dentro de cada fase.

P

Perfil de usuario: contiene los datos del usuario y los cursos que él cursará.

R

Recurso didáctico: Todos aquellos materiales con un diseño didáctico previo que apoyan el proceso de enseñanza aprendizaje y que contribuyen al desarrollo de la competencia comunicativa en el idioma inglés, en correspondencia con una concepción pedagógica definida.

Ruta de Autoaprendizaje: Se hace una prueba inicial para determinar si realmente el usuario está en el nivel que asegura y se proponen además una serie de ejercicios para darle la oportunidad de rectificar o recordar algo olvidado. Cada dato genera un perfil y cada perfil una ruta. Dichas rutas se comportan como un proceso cíclico por lo que deben permitir al usuario ir hacia delante o hacia atrás según vaya avanzando o no en el aprendizaje.

Registro de categoría: Archiva el resultado del usuario después de realizar la encuesta.

Recepcionista del CASIE: Se encarga de registrar todos los datos de los usuarios que visiten los CASIE, así como crear los contratos cuando los usuarios solicitan el Autoaprendizaje.

Registro de usuario: Contiene los datos del usuario y todas las operaciones que realiza en el sistema, es decir es donde se guarda toda la información referente al usuario.

T

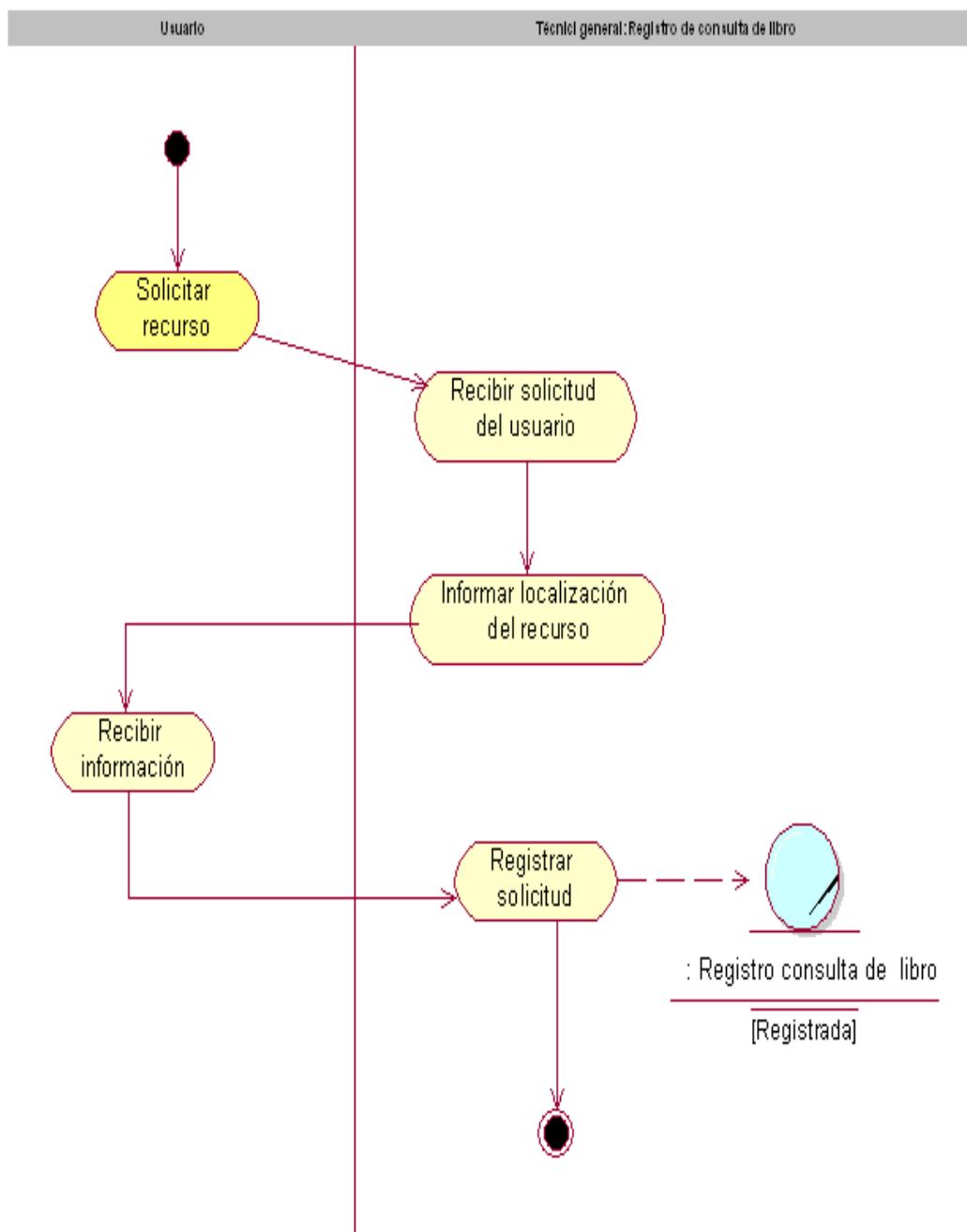
Técnico general del CASIE: Personal encargado de aplicar la encuesta a los usuarios una vez creado el contrato, así como registrar los datos de cuando se utilice un material para el Autoaprendizaje.

Técnico informático del CASIE: Personal encargado de instalar el software que el usuario necesite para su autoaprendizaje.

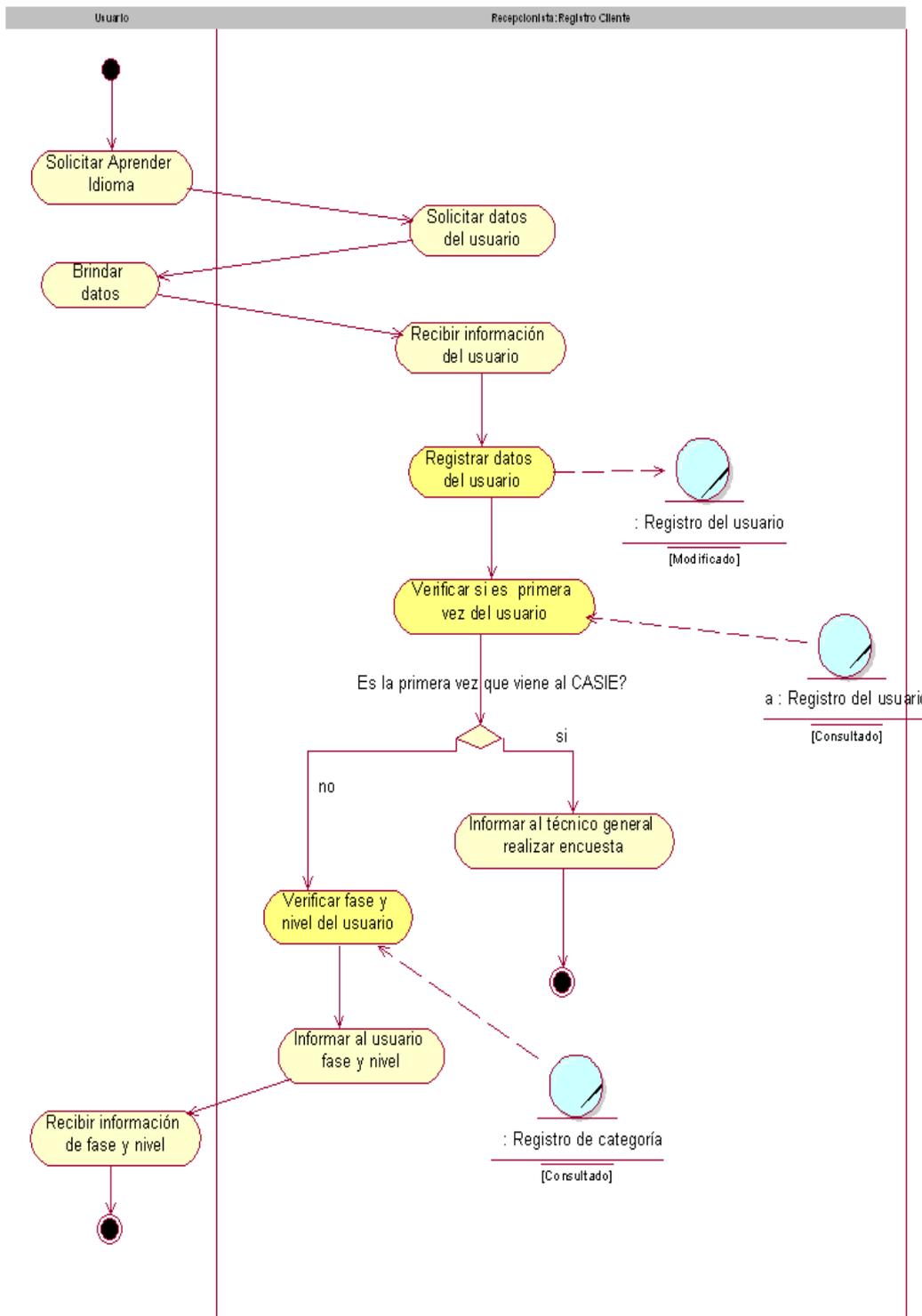
U

Usuario del CASIE: Los usuarios del CASIE son aquellas personas (estudiantes, trabajadores docentes o no docentes) que solicitan una ruta de autoaprendizaje para el estudio de las lenguas extranjeras.

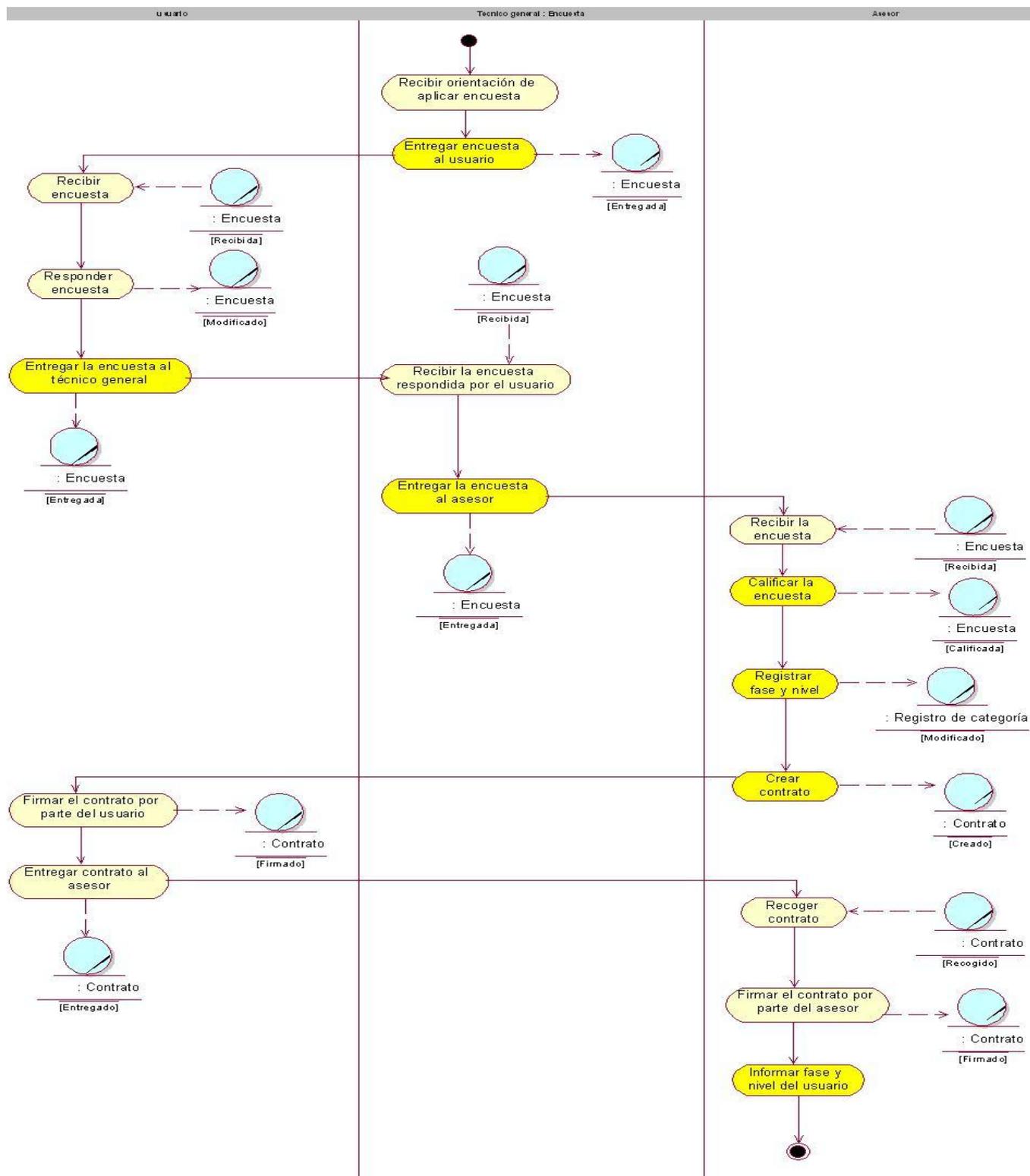
Anexos



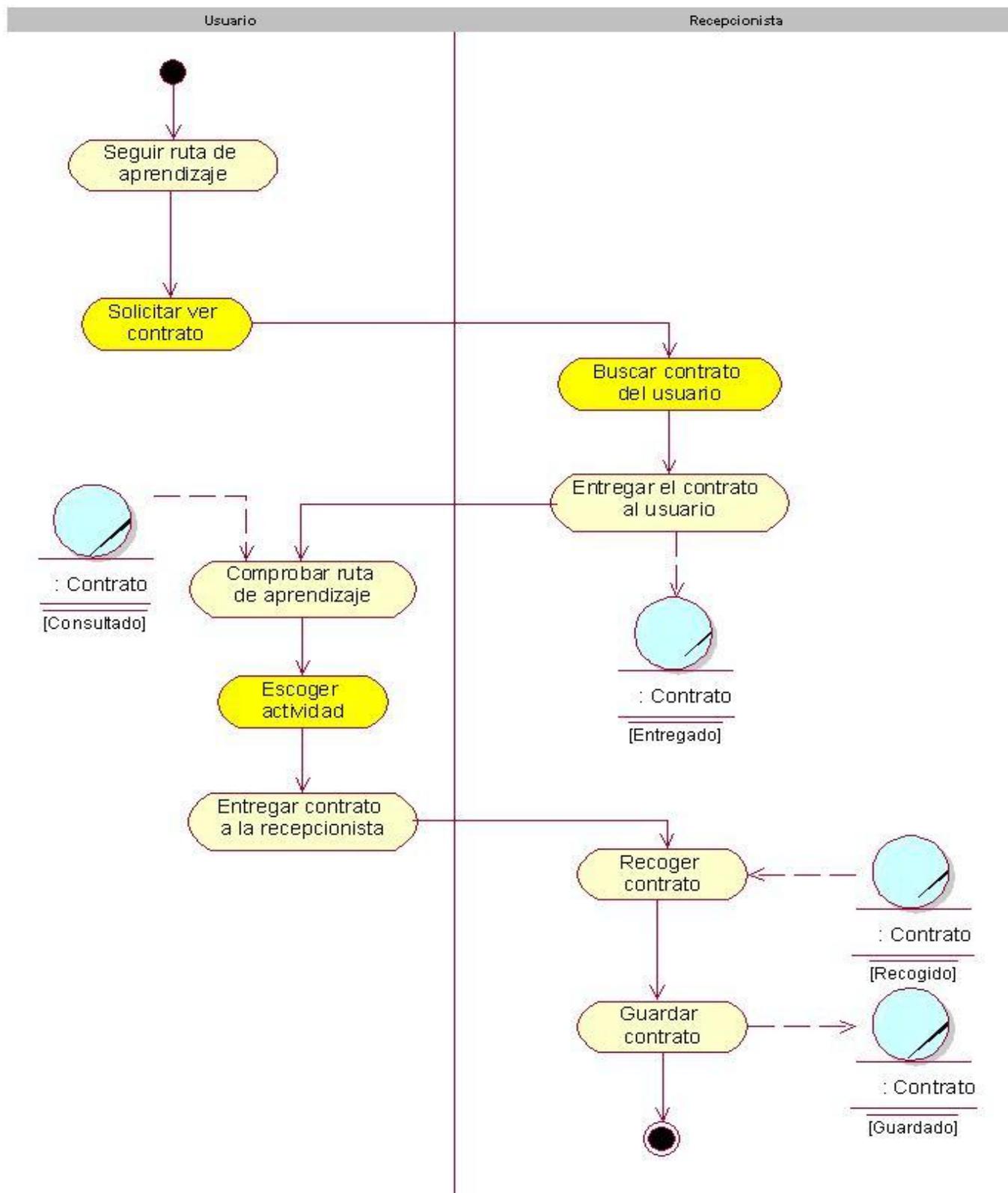
Anexo 1. Diagrama de Actividades del CUN_Solicitar Recurso



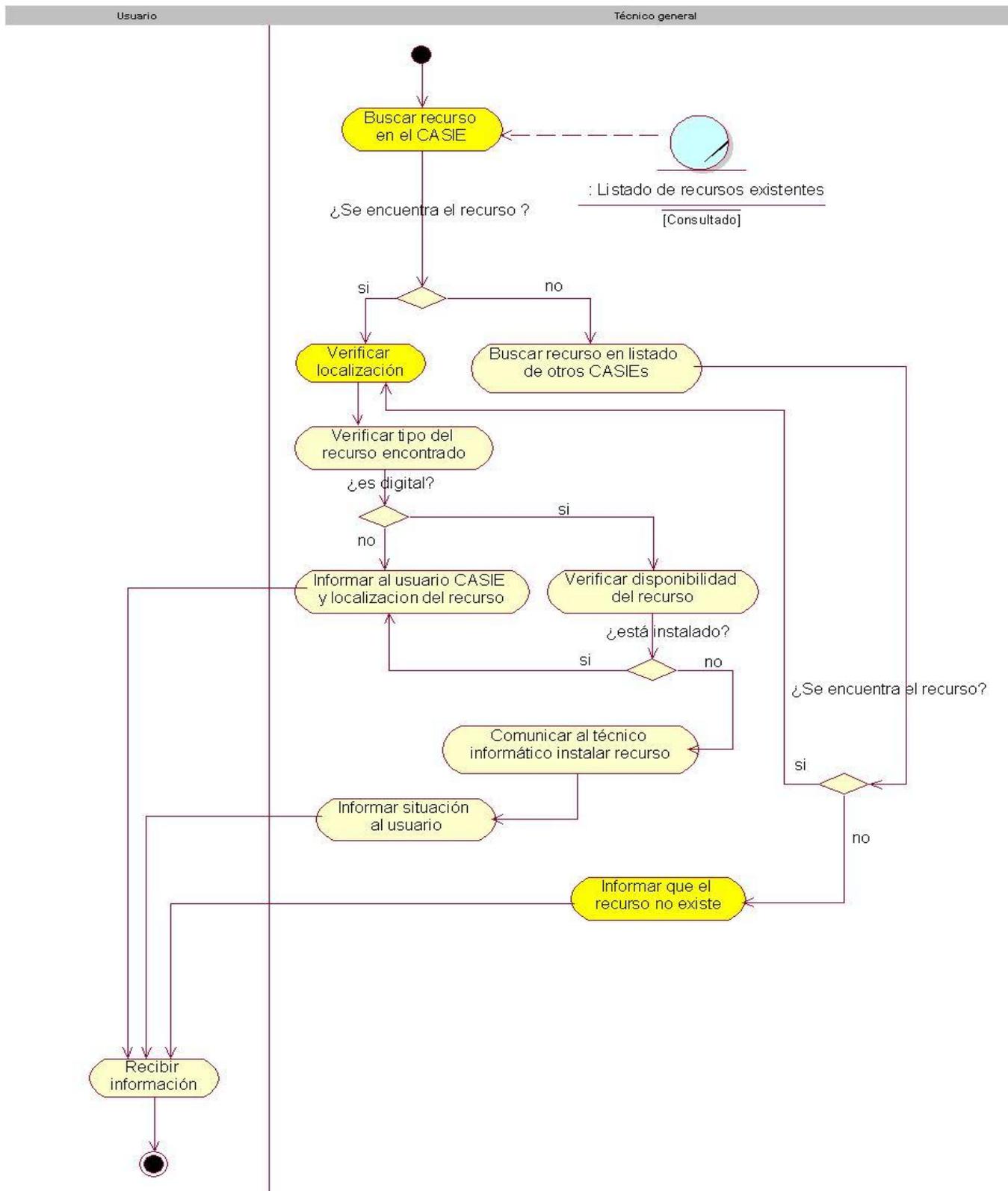
Anexo 2. Diagrama de Actividades del CUN_Atender usuario



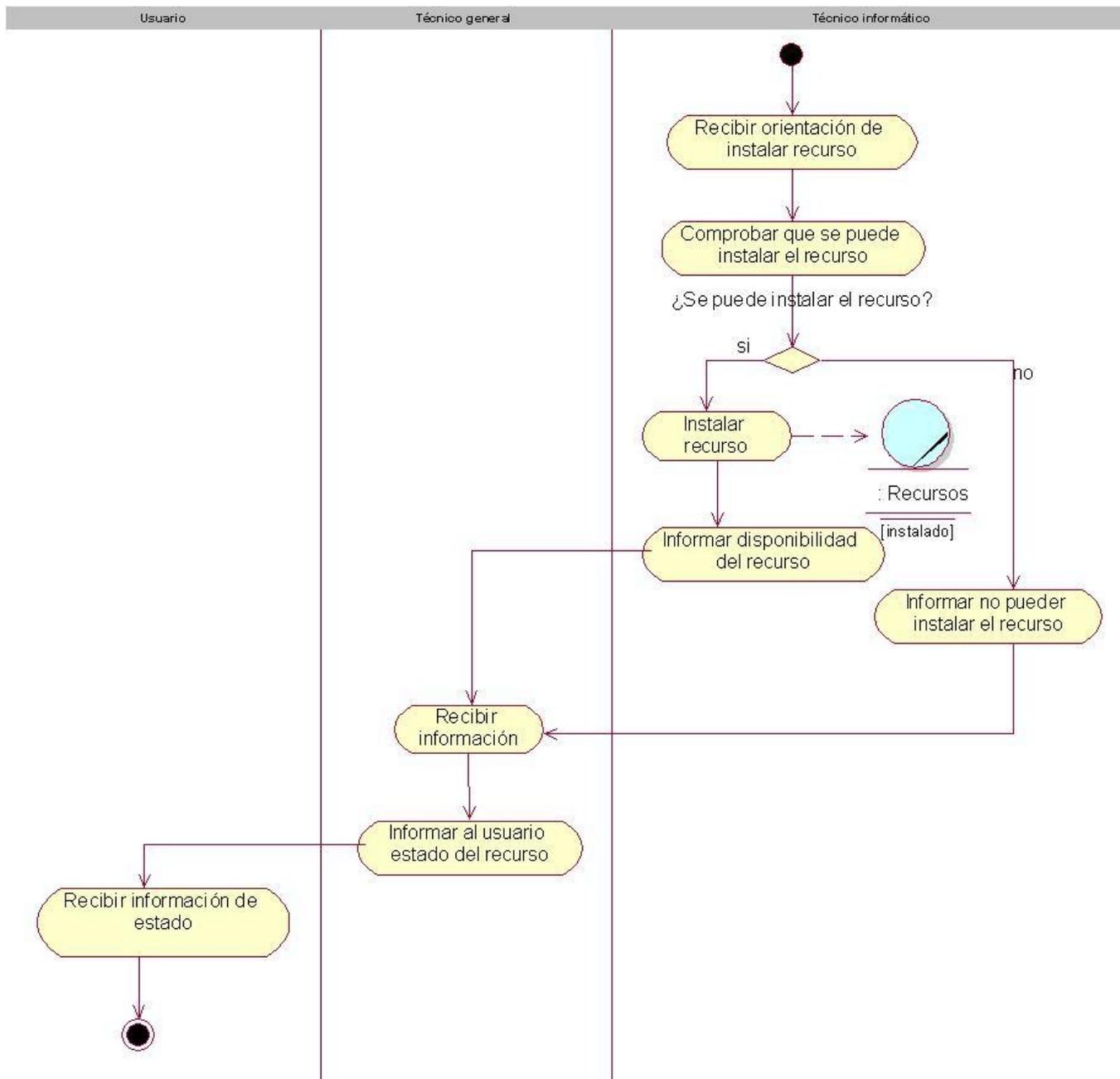
Anexo 3. Diagrama de Actividades del CUN_Realizar Encuesta



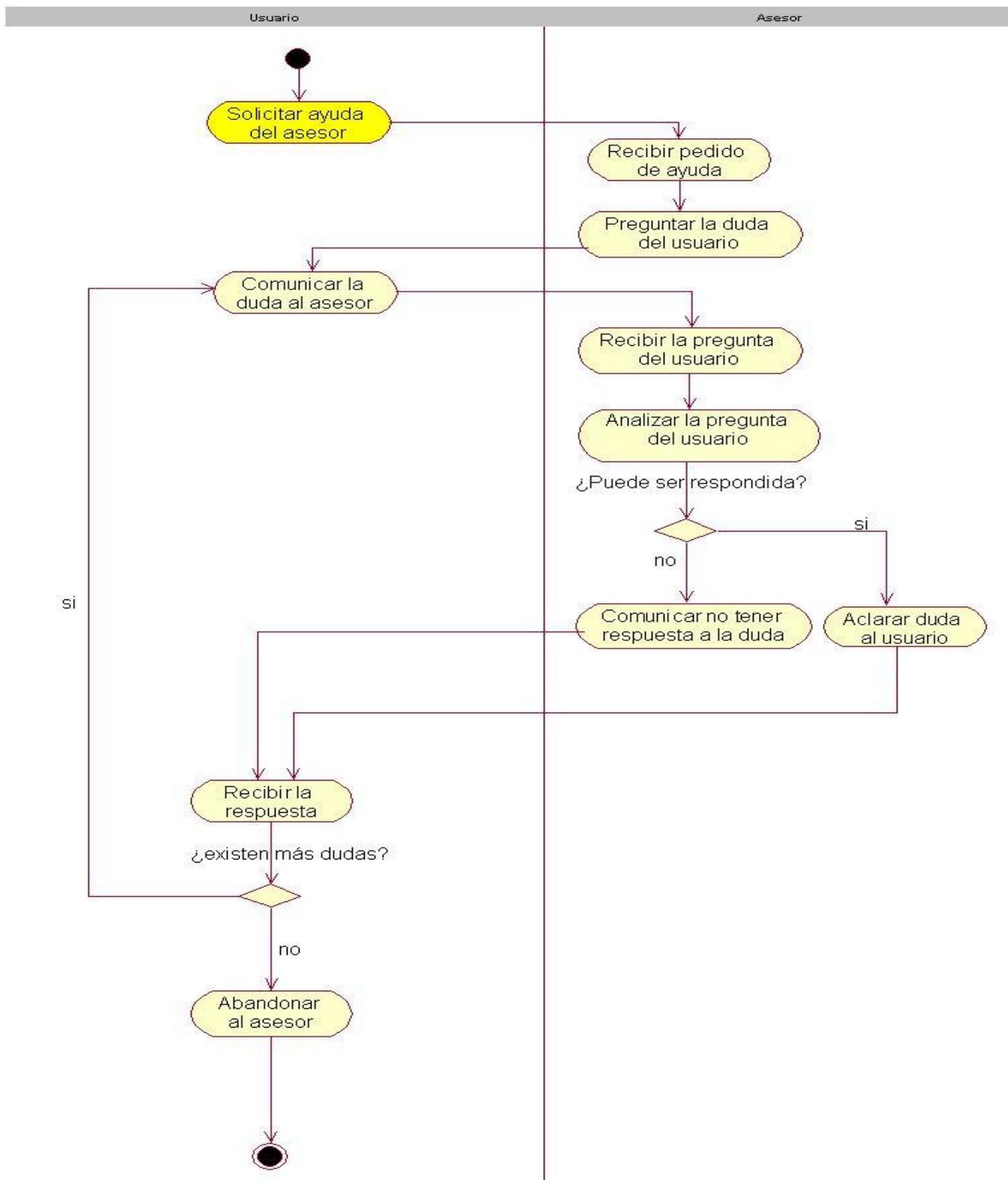
Anexo 4. Diagrama de Actividades del CUN_Seguir Ruta de aprendizaje



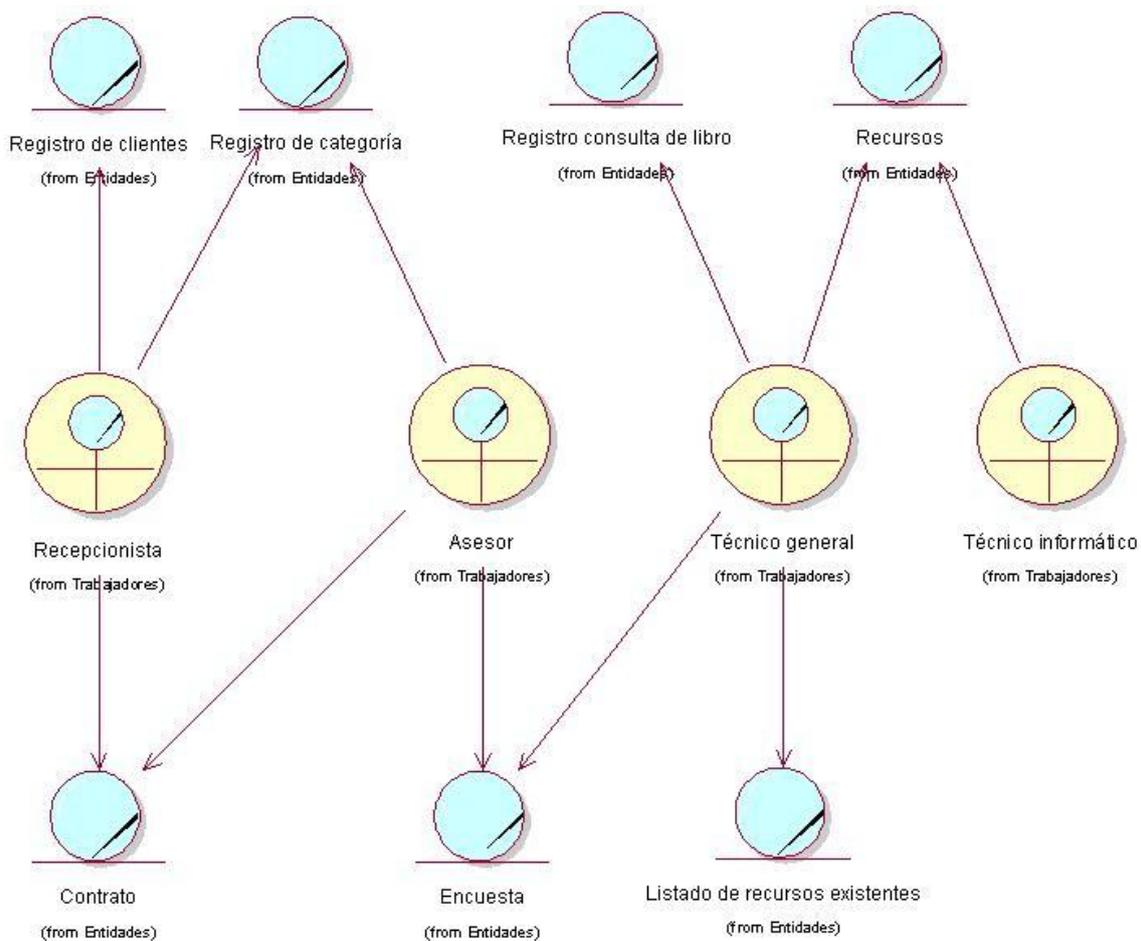
Anexo 5. Diagrama de Actividades del CUN_Localizar Recurso



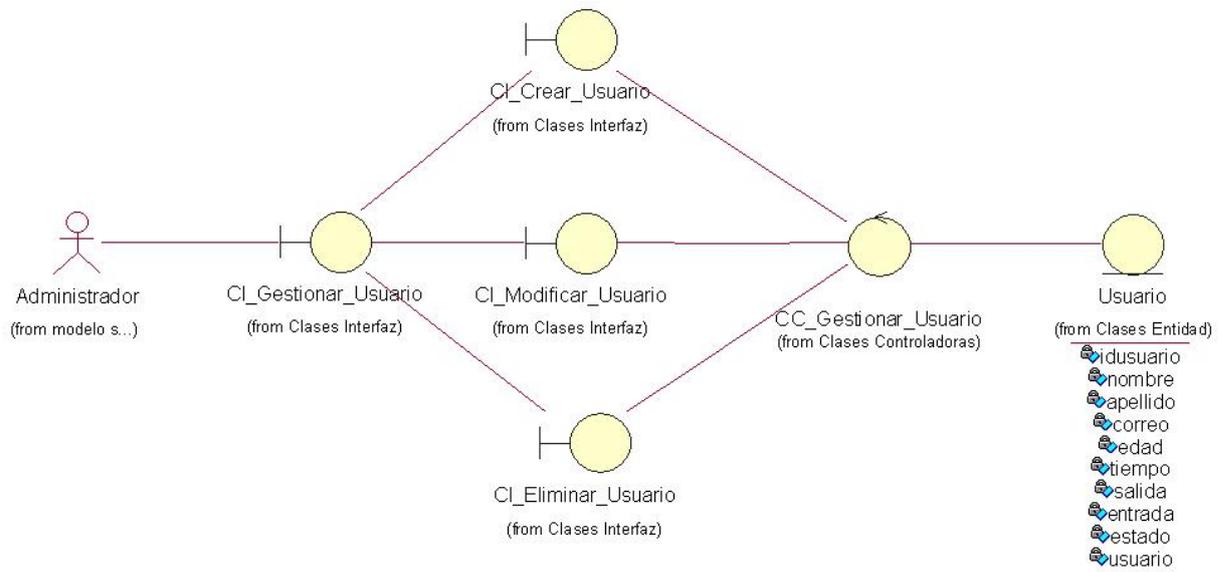
Anexo 6. Diagrama de Actividades del CUN_Instalar Recurso



Anexo 7. Diagrama de Actividades del CUN_Solicitar Asesoría



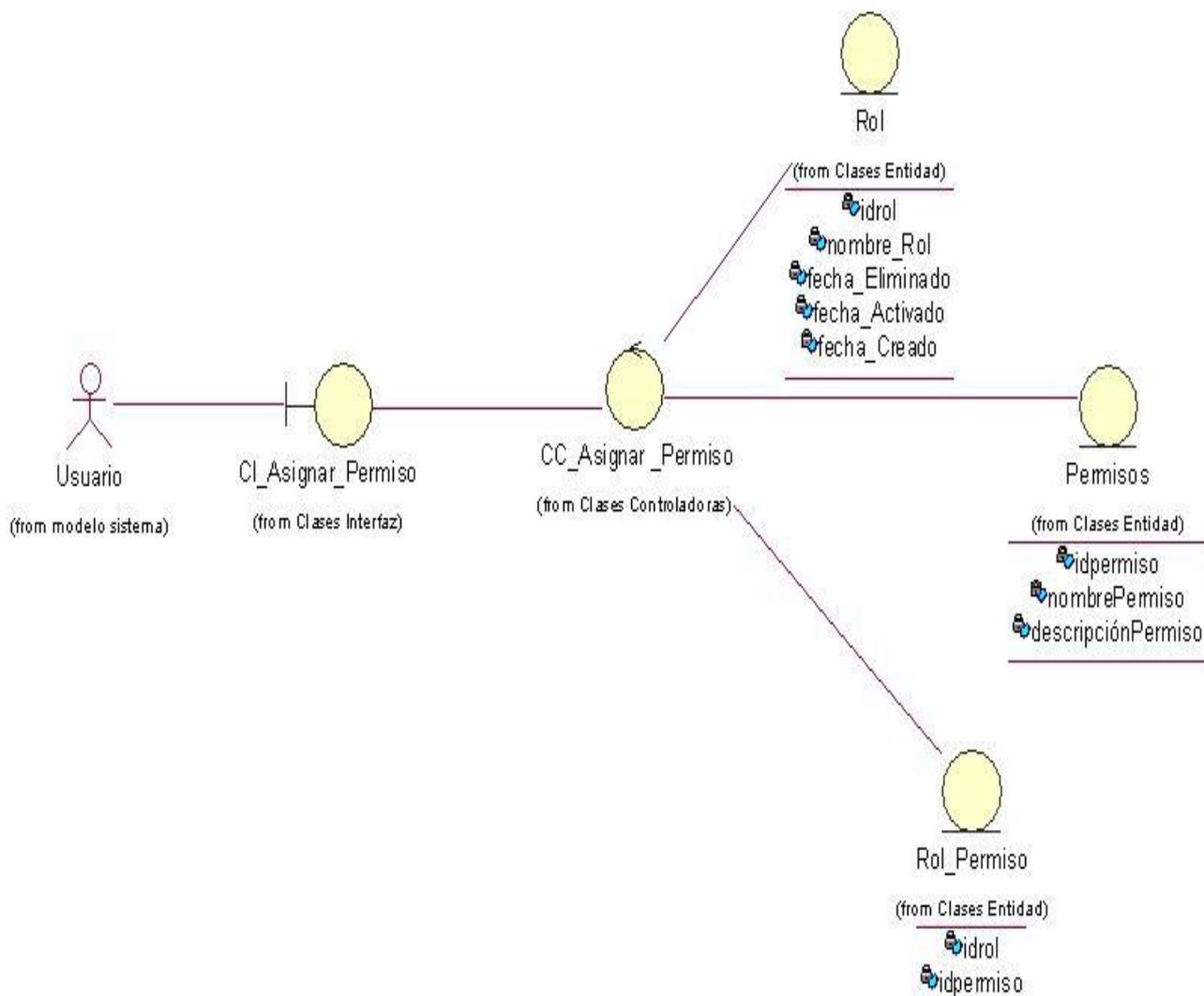
Anexo 8. Modelo de Objeto del Negocio



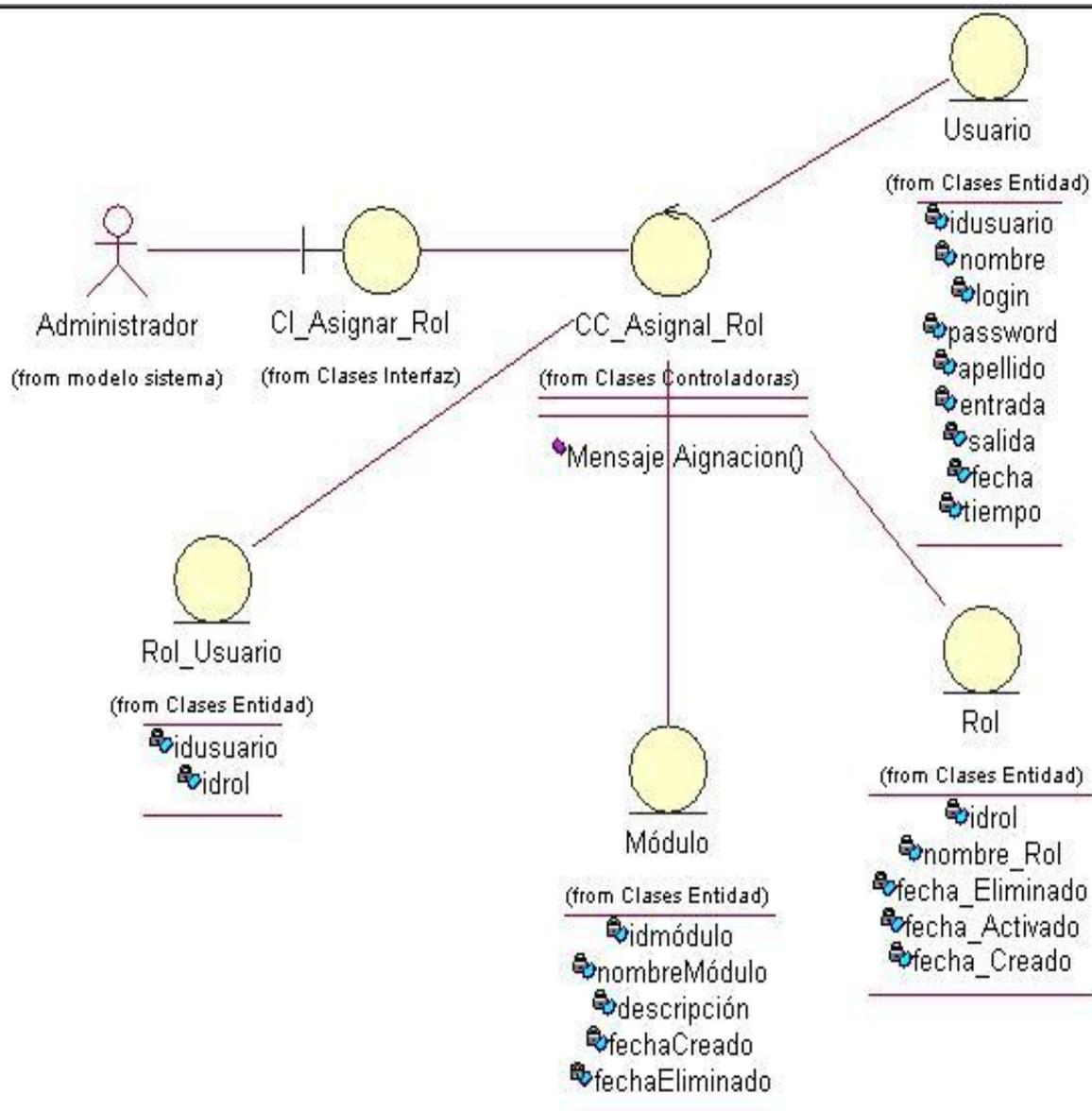
Anexo 9. Diagrama de clases del análisis del CUS_Gestionar Usuario



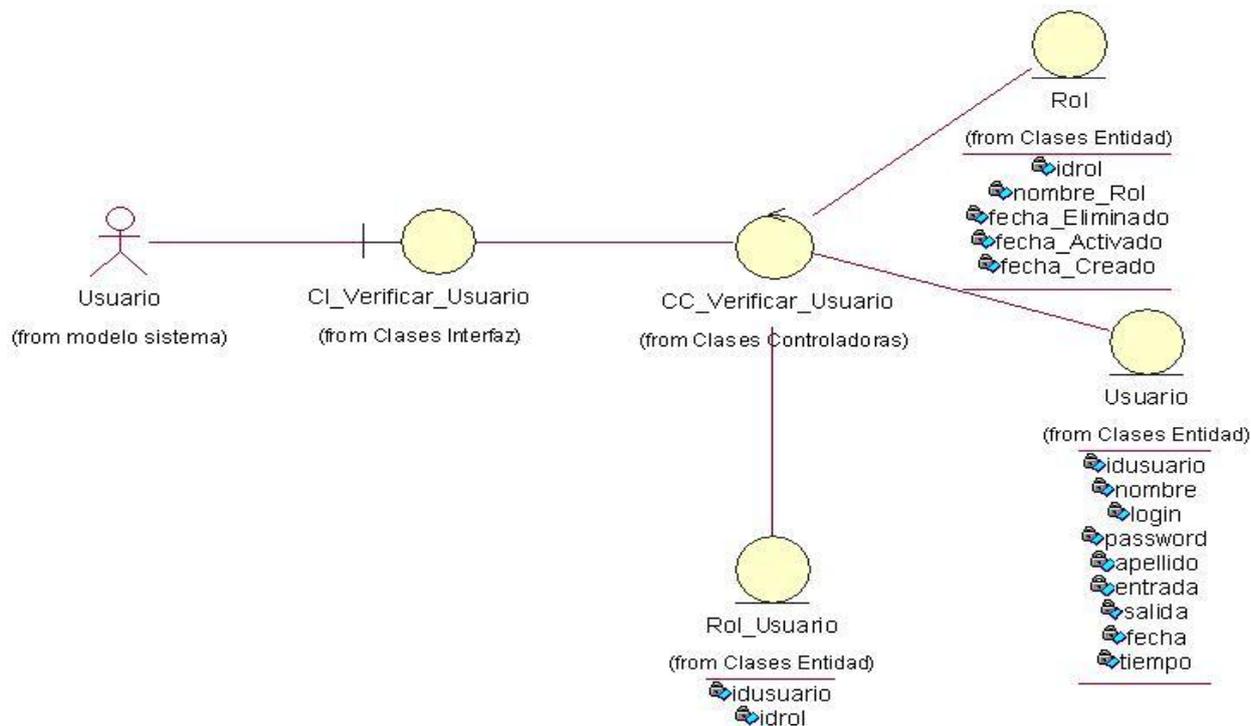
Anexo 10 Diagrama de clases del análisis del CUS_Construir Perfil



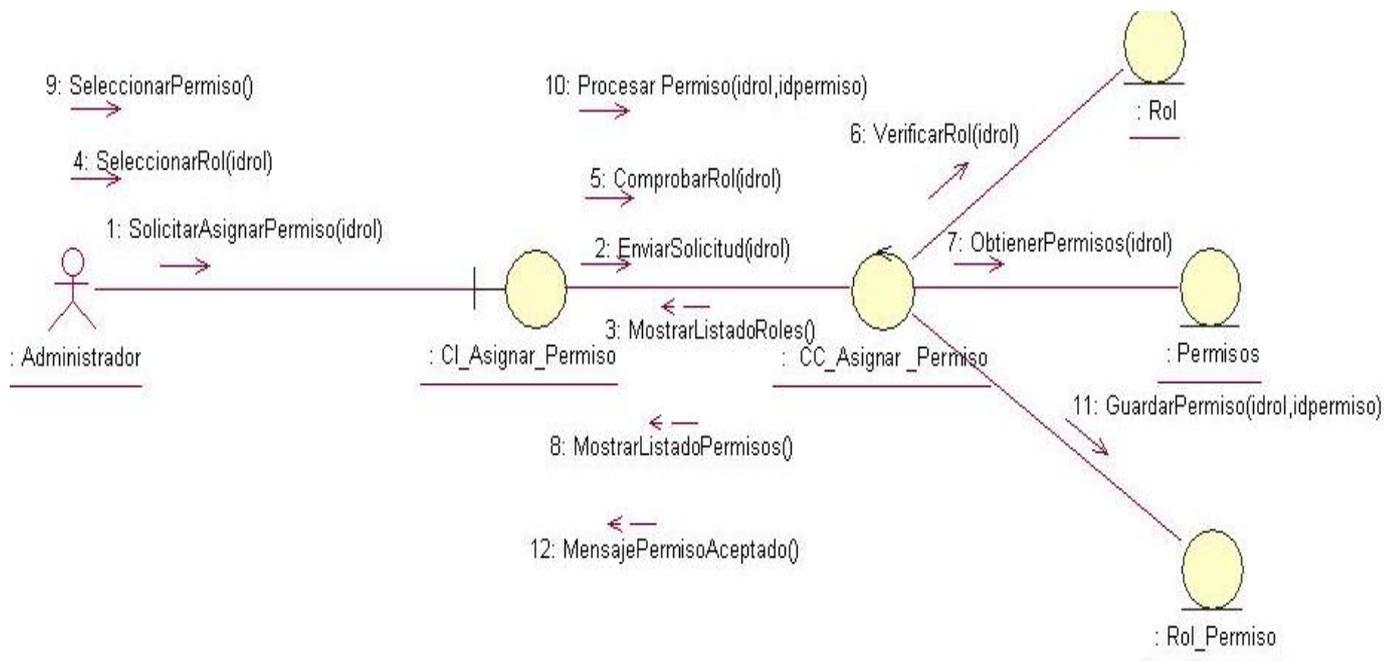
Anexo 11 Diagrama de clases del análisis del CUS_Asignar Permisos



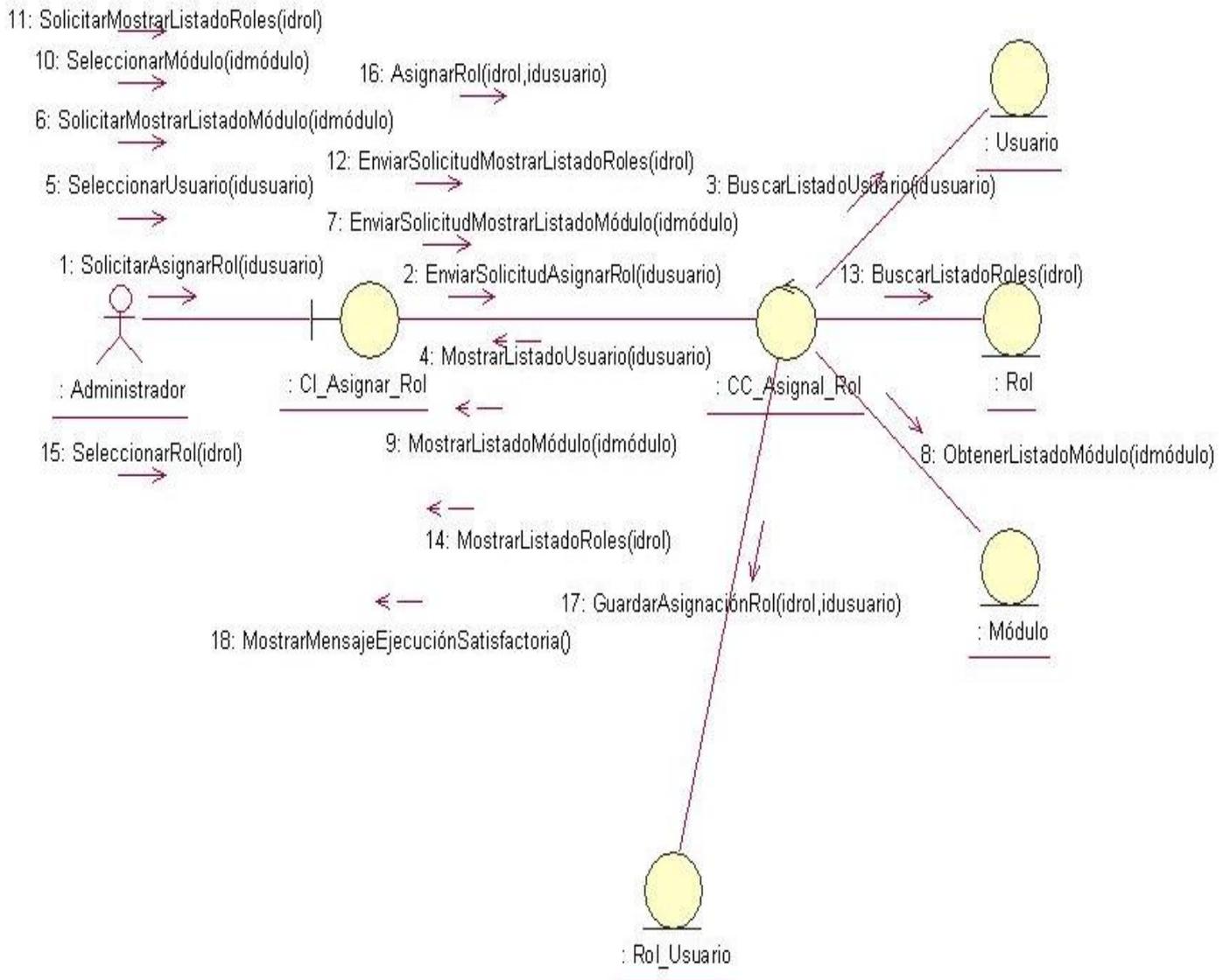
Anexo 12 Diagrama de clases del análisis del CUS_Asignar Rol



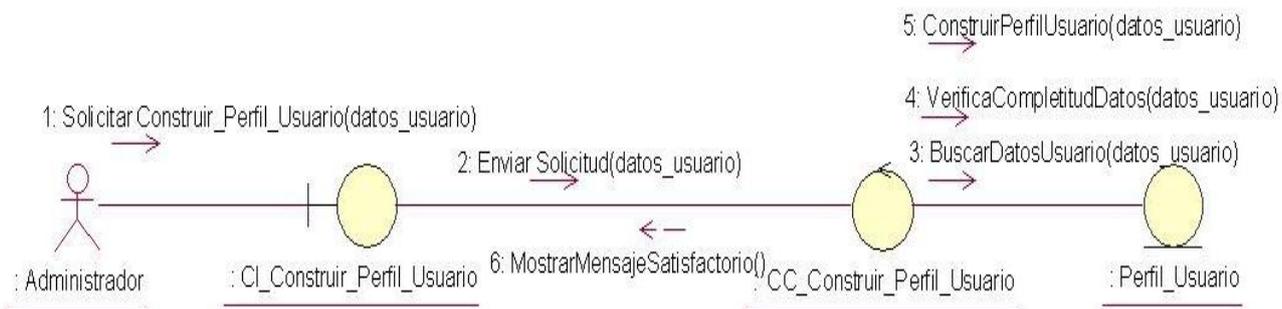
Anexo 13 Diagrama de clases del análisis del CUS_Verificar Usuario



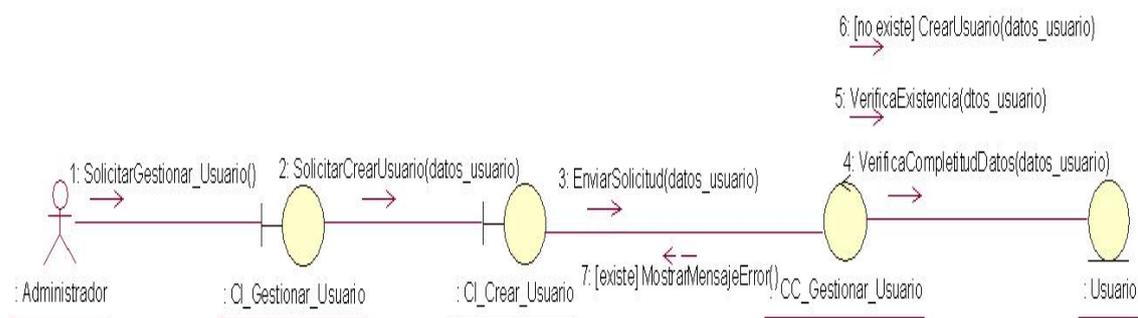
Anexo 14. Diagrama de Colaboración de Asignar Permiso



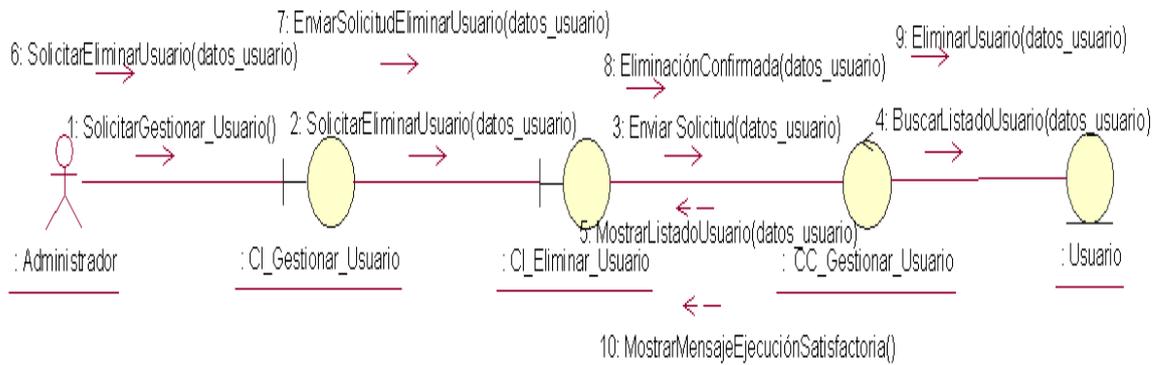
Anexo 15 Diagrama de Colaboración del CUS_Asignar Rol



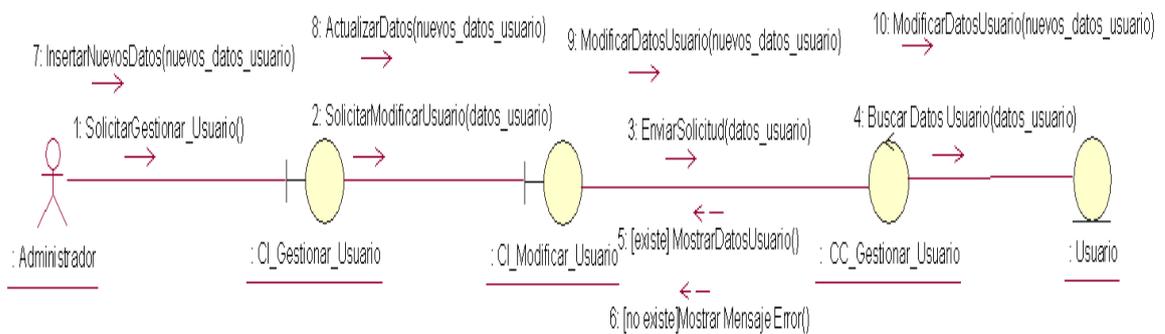
Anexo 16 Diagrama de Colaboración del CUS_Construir Perfil de usuario



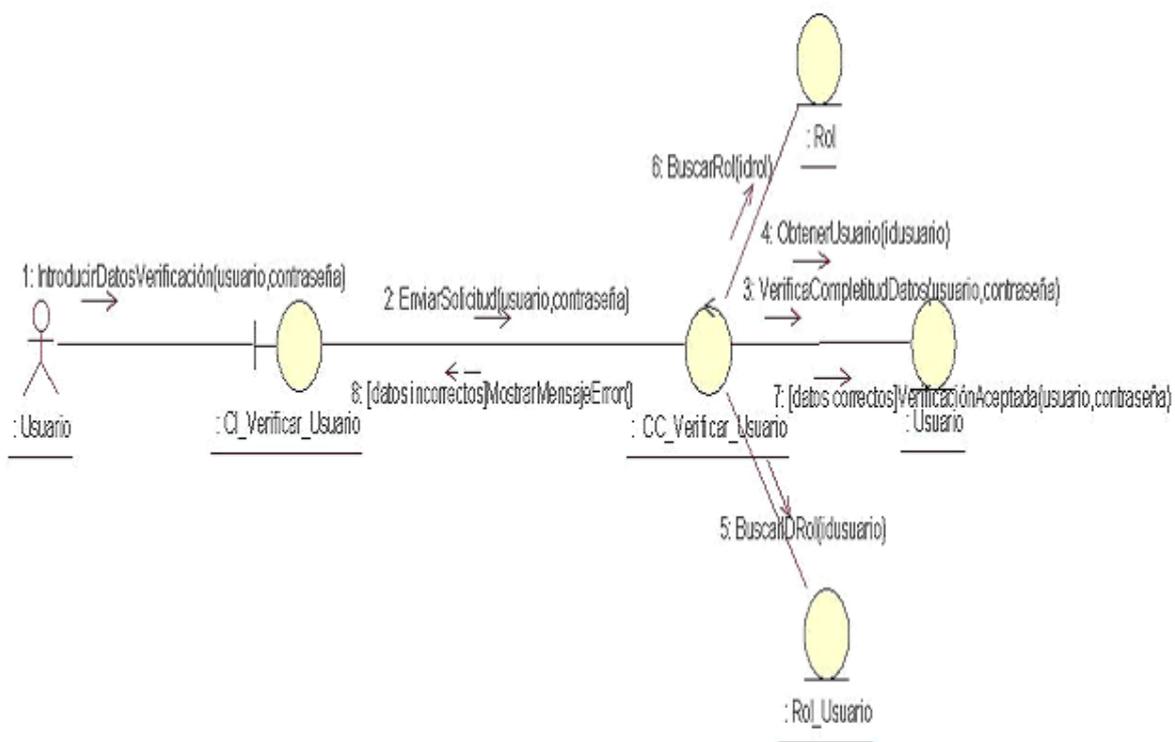
Anexo 17 Diagrama de Colaboración del escenario Crear Usuario



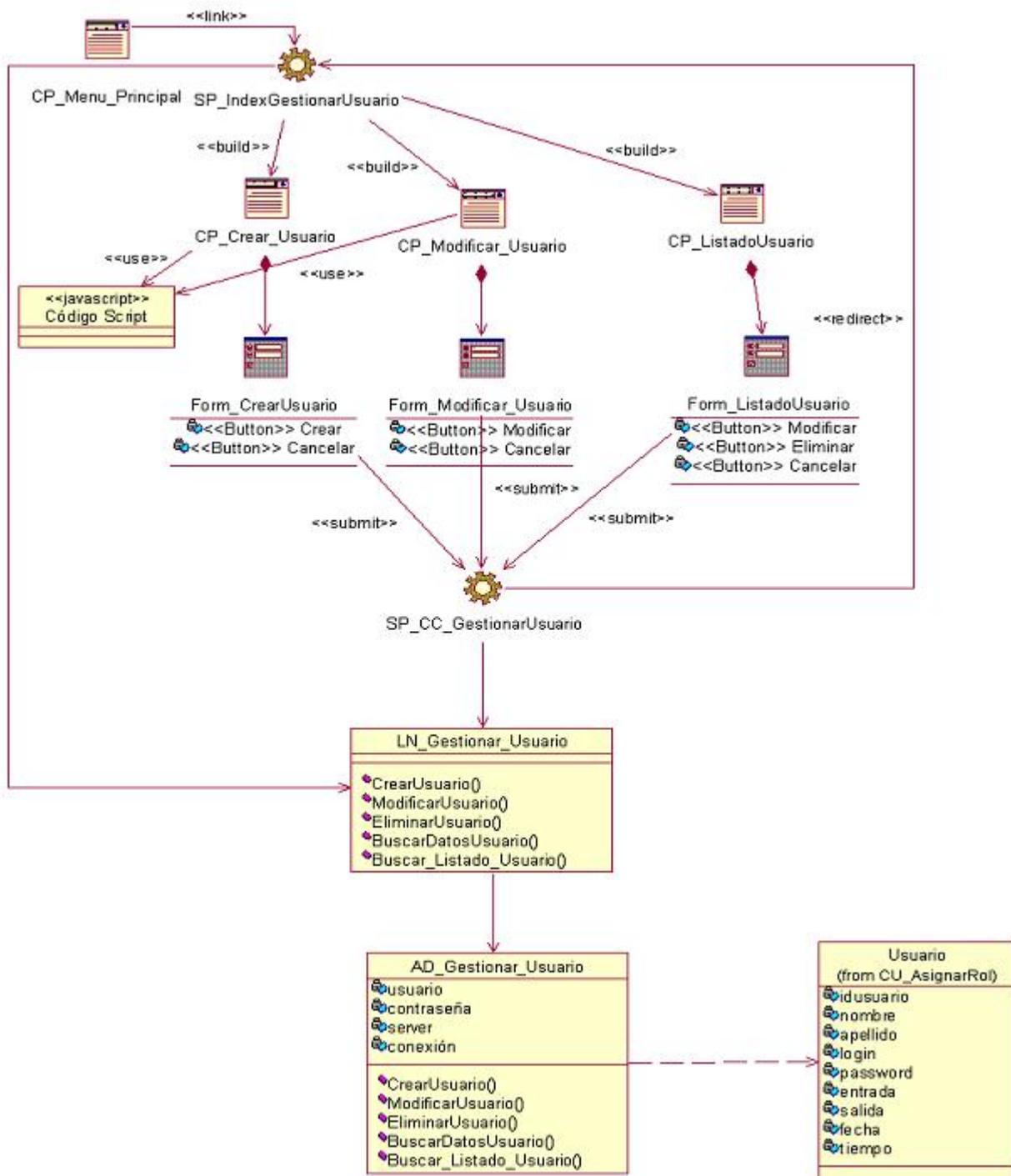
Anexo 17.1 Diagrama de Colaboración del escenario Eliminar Usuario



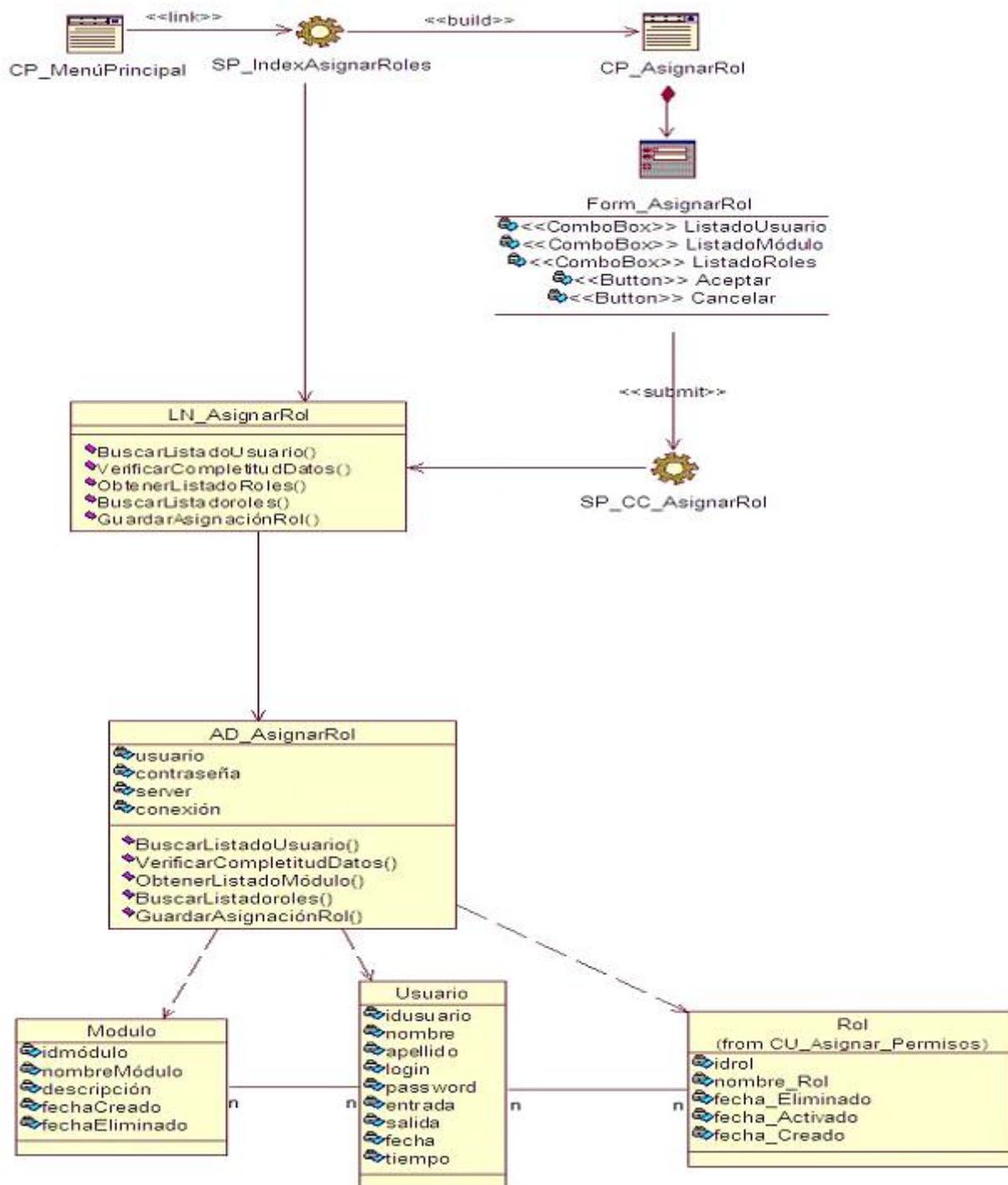
Anexo 17.2 Diagrama de Colaboración del escenario Modificar Usuario



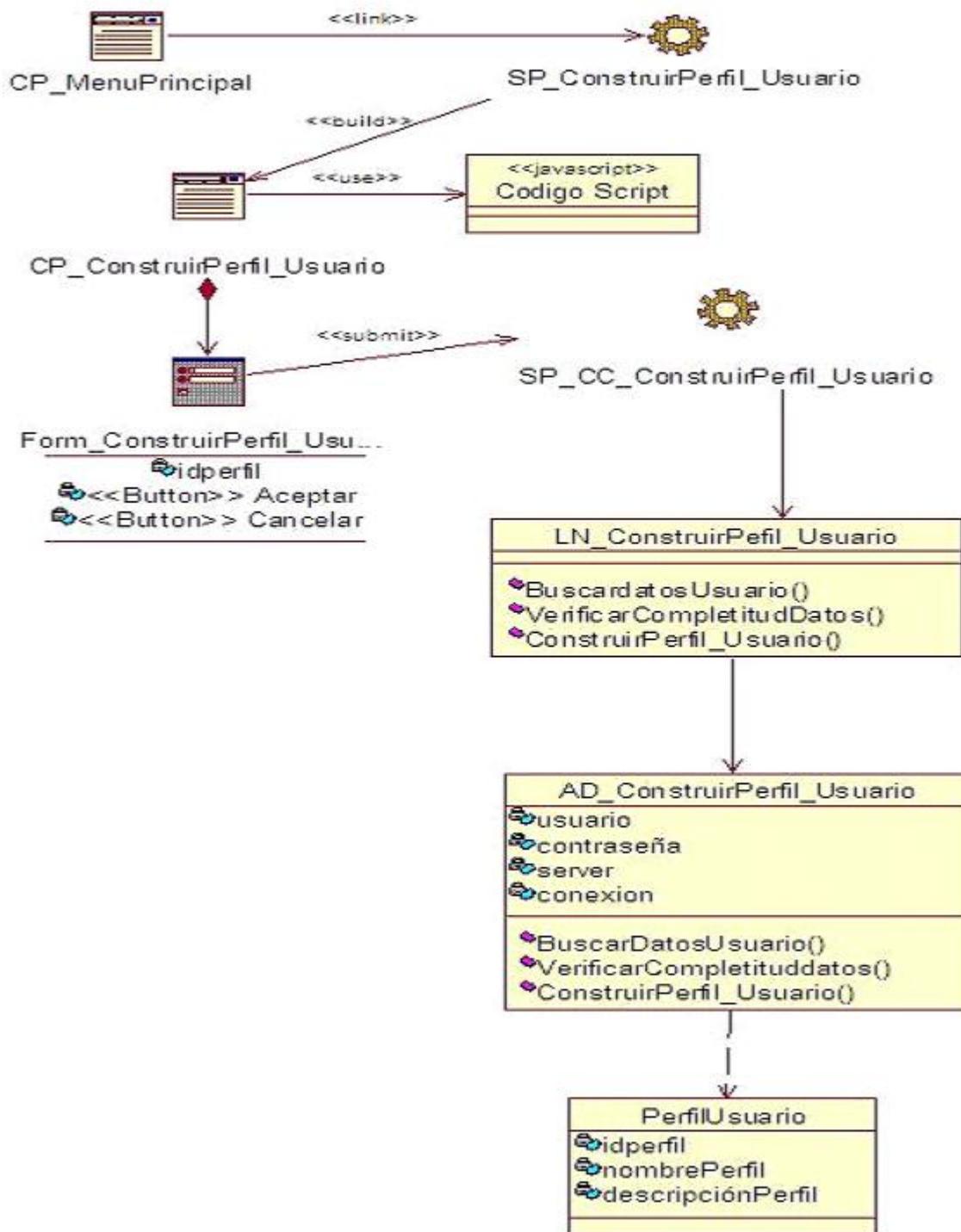
Anexo 18 Diagrama de Colaboración del CUS_Verificar Usuario



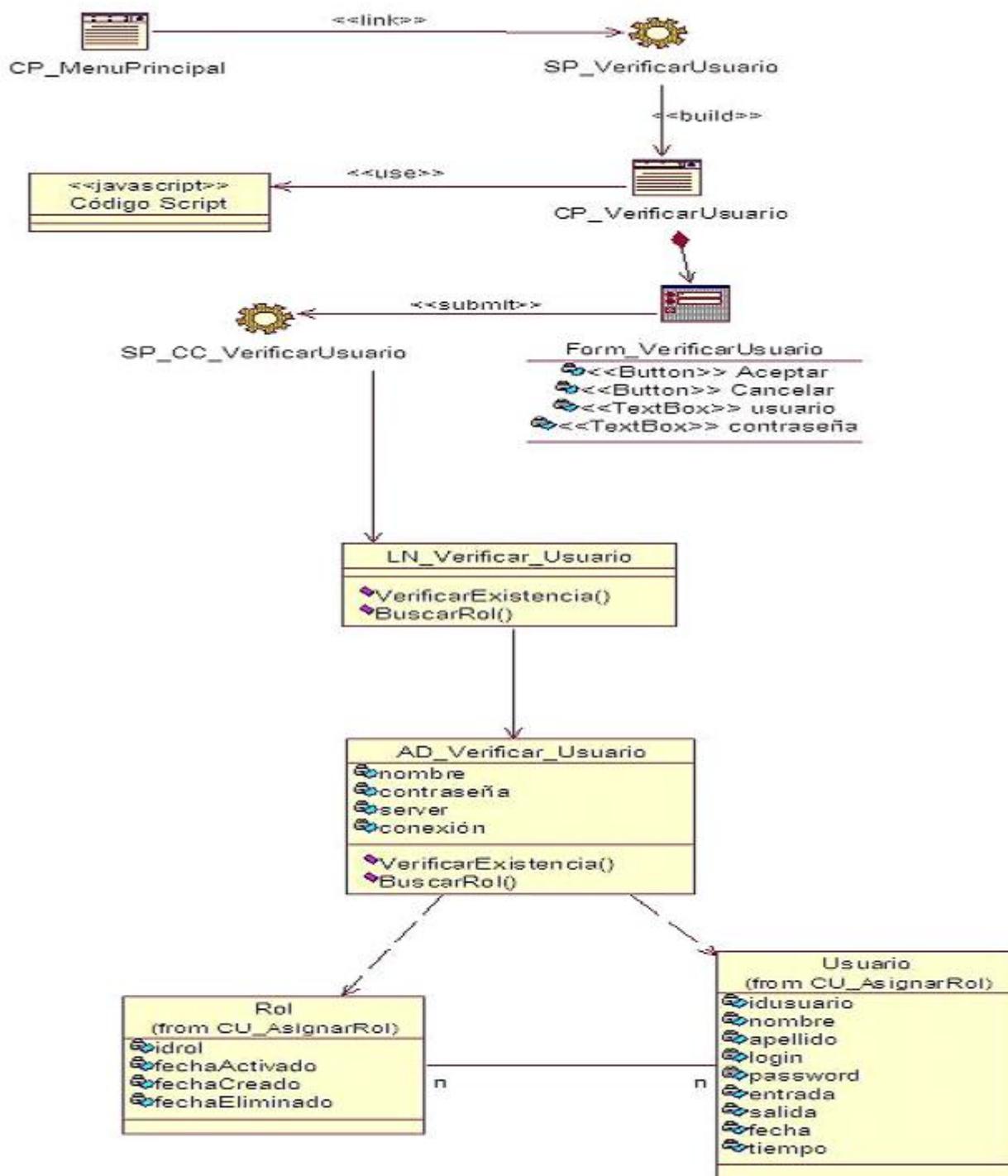
Anexo 19 Diagrama de clases de diseño del CUS_Gestionar Usuario



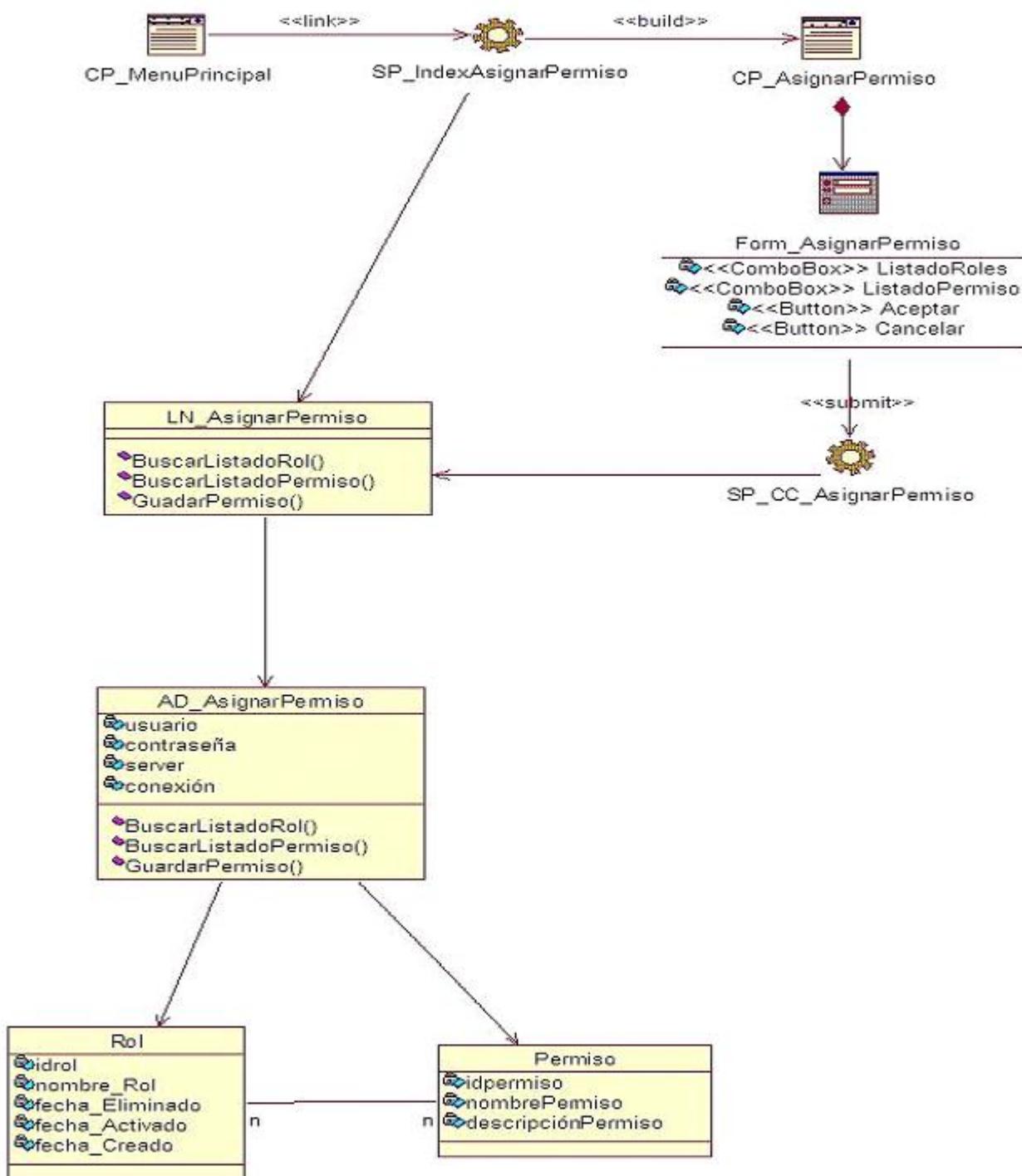
Anexo 20 Diagrama de clases de diseño del CUS_Asignar rol



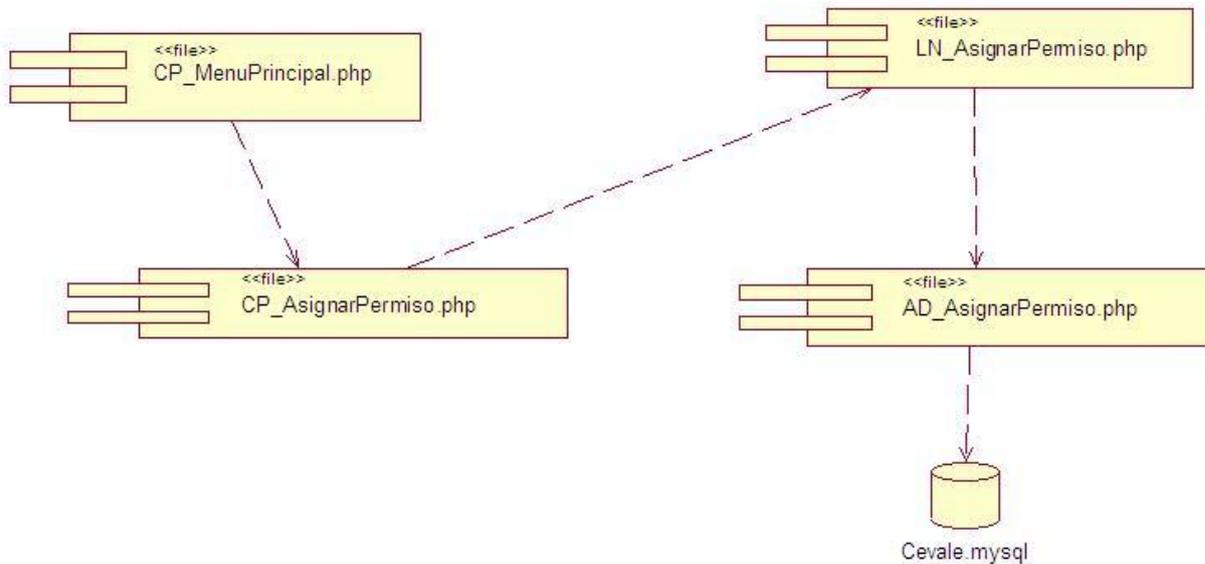
Anexo 21 Diagrama de clases de diseño del CUS_Construir perfil de usuario



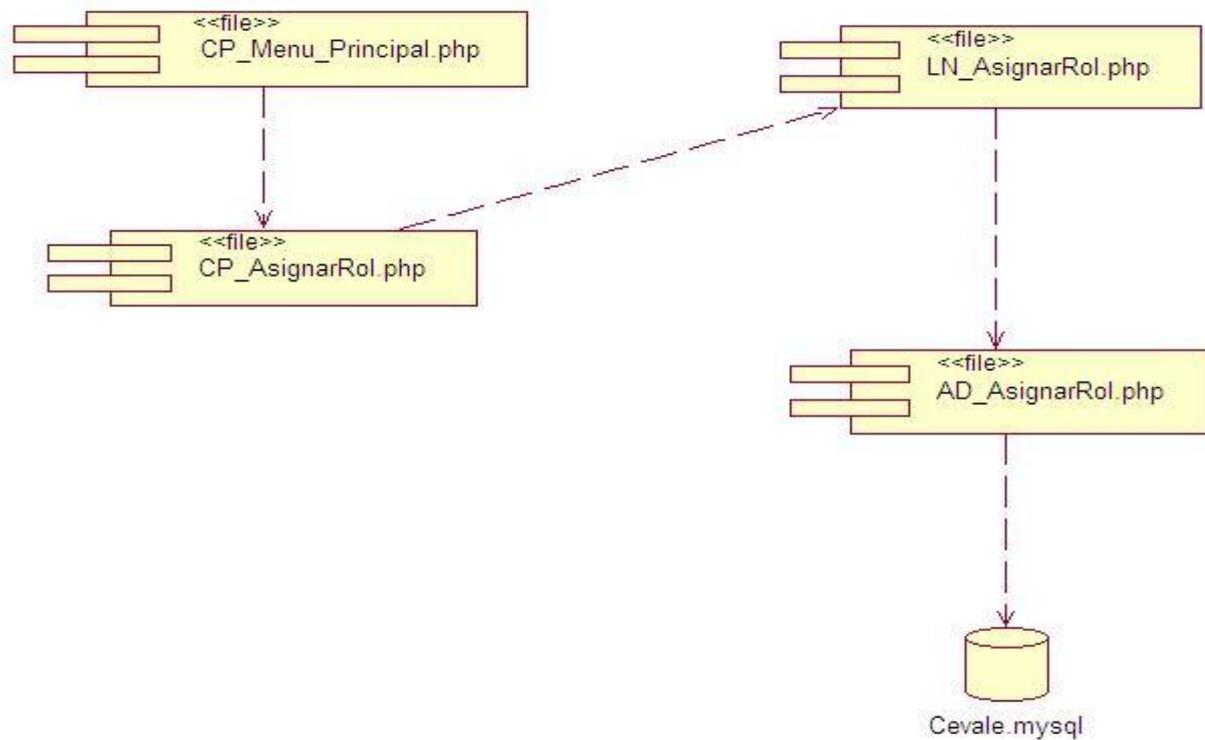
Anexo 22 Diagrama de clases de diseño del CUS_Verificar usuario



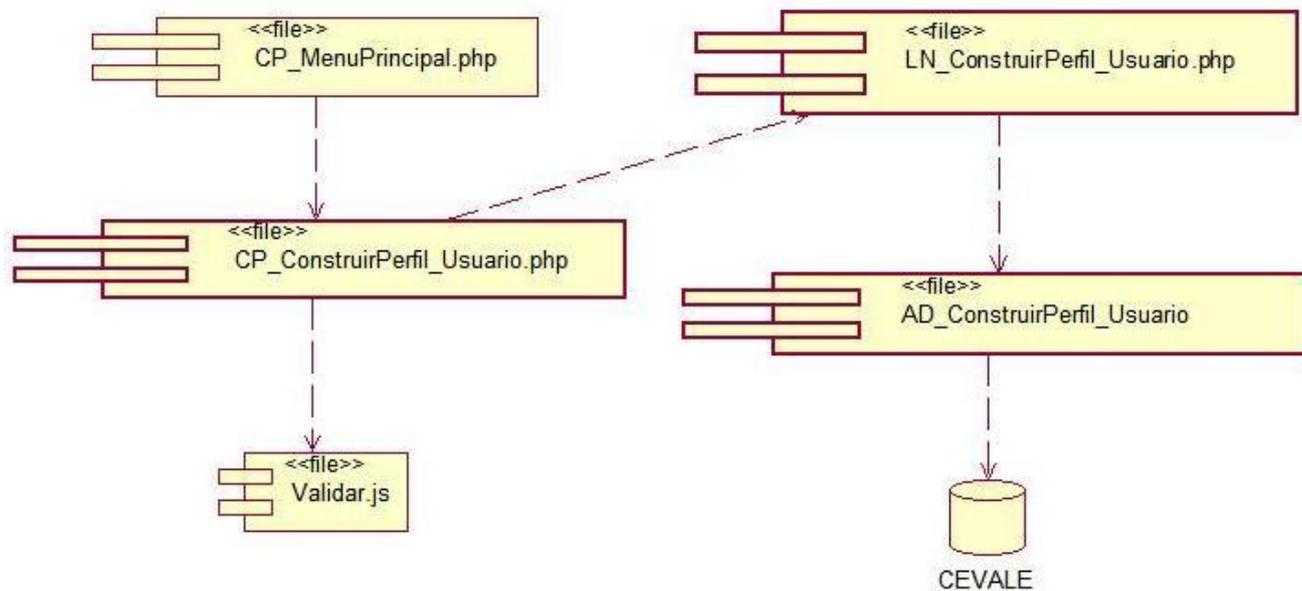
Anexo 23 Diagrama de clases de diseño del CUS_Asignar Permiso



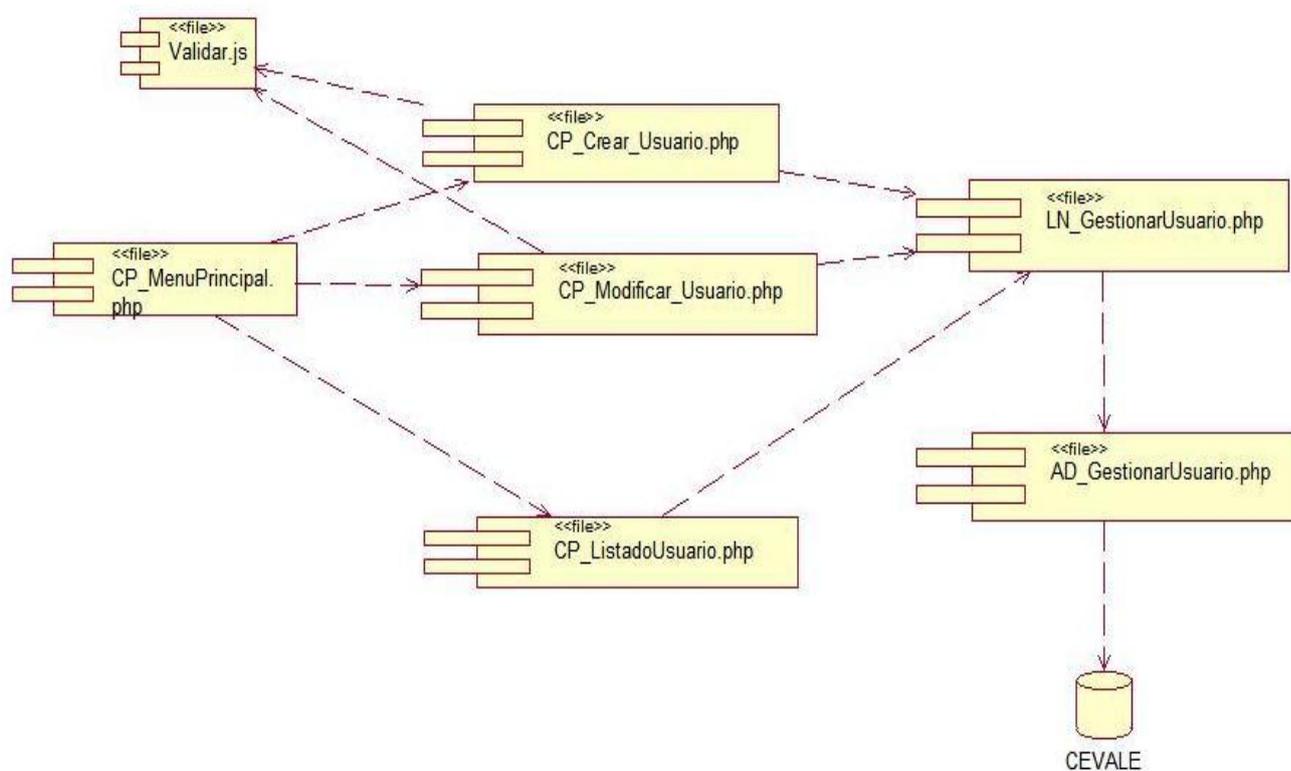
Anexo 24 Diagrama de Componentes del CUS_Asignar Permiso



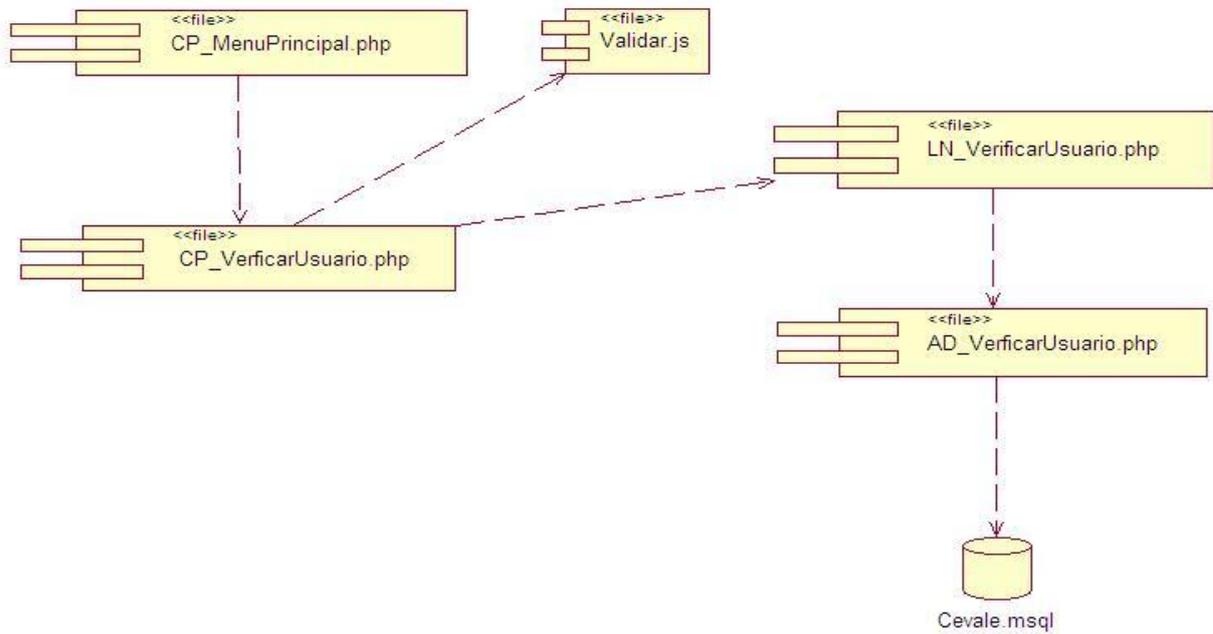
Anexo 25 Diagrama de Componentes del CUS_Asignar Rol



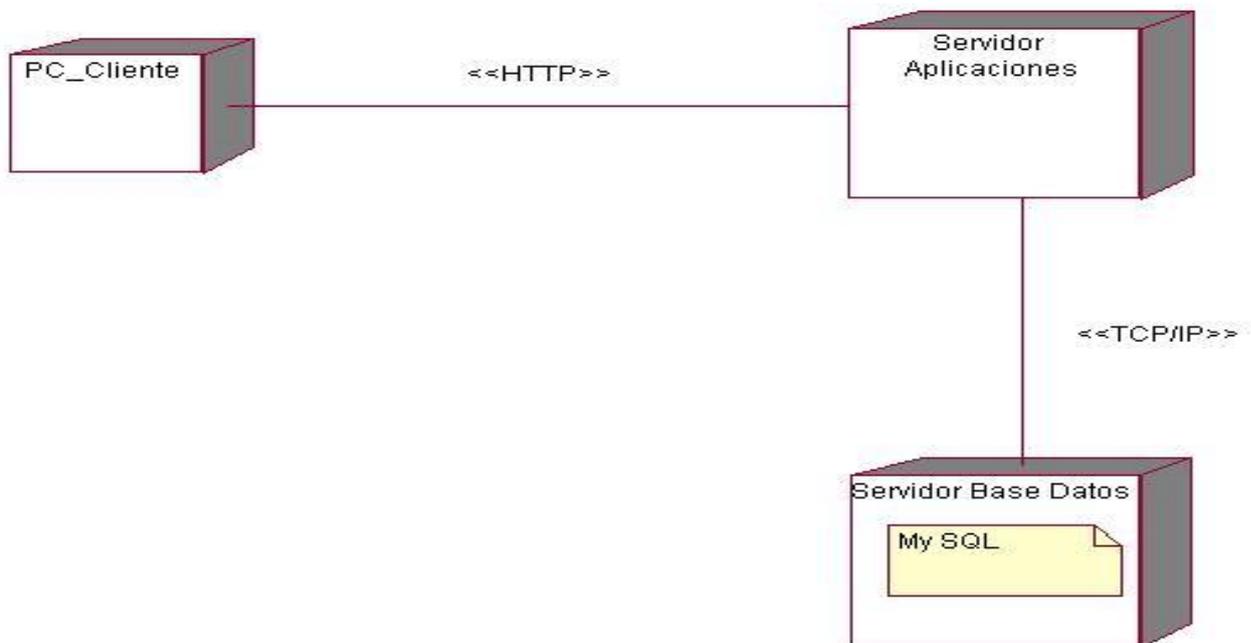
Anexo 26 Diagrama de Componentes del CUS_Construir Perfil de Usuario



Anexo 27 Diagrama de Componentes del CUS_Gestionar Usuario



Anexo 28 Diagrama de Componentes del CUS_Verificar Usuario



Anexo 29 Modelo de Despliegue

Proyecto CEVALE
Módulo de Administración

MODELO DE ENTREVISTA

Entrevistado: _____

Entrevistador: _____

Fecha: _____

Cuestionario

1. ¿Cuáles son las técnicas de autenticación utilizadas en la UCI?
2. ¿Cuál es la que más se utiliza?
3. ¿Qué nivel de seguridad tienen estas técnicas?
4. ¿Qué longitud debe tener la contraseña?
5. ¿Qué composición debe tener la contraseña?
6. ¿Cada qué tiempo se cambia la contraseña?
7. ¿Qué método de encriptación se usa, o cuáles usted aconseja?
8. Principales ataques informáticos que ha sufrido la UCI, en cuanto a los diferentes métodos de autenticación.
9. ¿Qué medidas de seguridad se usan aquí en la UCI?