

CONTENIDO

CONTENIDO	1
RESUMEN.....	4
1.- MARCO TEÓRICO.....	5
1.1.- SISTEMAS DE COLABORACIÓN.....	5
1.1.1.- INTRODUCCIÓN.....	5
1.1.2.- DEFINICIONES	6
1.1.2.1.- Conocimiento	6
1.1.2.2.- Colaboración.....	6
1.1.2.3.- Groupware	6
1.1.3.- CARACTERÍSTICAS	7
1.1.3.1.- Aplicación del Groupware.....	7
1.2.- HERRAMIENTAS OPEN SOURCE	8
1.2.1.- INTRODUCCIÓN.....	8
1.2.2.- VENTAJAS DEL OPEN SOURCE.....	9
1.2.2.1.- Administración	9
1.2.2.2.- Seguridad	9
1.2.2.3.- Costos	9
1.2.3.- REGULACIÓN DE LAS LICENCIAS.....	10
1.2.4.- LOS RÉDITOS DEL OPEN SOURCE.....	11
1.3.- SISTEMAS DE MENSAJERÍA.....	11
1.3.1.- CONCEPTOS	11
1.3.2.- ESTADÍSTICAS DE USO DE LOS SISTEMAS DE MENSAJERÍA INSTANTÁNEA	13
1.4.- METODOLOGÍA DE DESARROLLO.....	14
1.4.1.- INTRODUCCIÓN.....	14
1.4.3.- LOS PROCESOS DEL CICLO DE VIDA DEL SOFTWARE	15
1.4.4.- PROCESOS DEL DESARROLLO DE SOFTWARE.....	15
1.4.5.- PROCESO UNIFICADO DE DESARROLLO DEL SOFTWARE.....	16
1.4.5.1.- Historia.....	17

1.4.5.2.- Características del Proceso Unificado de Desarrollo de Software.....	19
1.4.5.3.- Casos de Uso	19
1.4.5.4.- Modelos de Casos y Usos	20
1.4.5.5.- El ciclo de vida del Proceso Unificado de Desarrollo.....	21
1.4.5.6.- Iterativo e Incremental	21
1.4.5.7.- Centrado en la Arquitectura	25
2.- ANÁLISIS Y DISEÑO DEL SISTEMA.....	26
2.1.- REQUERIMIENTOS	26
2.1.1.- PLANTEAMIENTO DEL PROBLEMA.....	26
2.1.2.- FORMULACIÓN DEL PROBLEMA	26
2.1.3.- OBJETIVO GENERAL DEL PROYECTO	26
2.1.4.- REQUISITOS FUNCIONALES Y NO FUNCIONALES.....	27
2.1.5.- MODELO DEL NEGOCIO	27
2.1.6.- MODELO CONCEPTUAL O DEL DOMINIO	28
2.1.7.- IDENTIFICACIÓN DE ACTORES Y CASOS DE USO	29
2.1.7.1.- Casos de Uso del Sistema	29
2.1.8.- MODELO DE CASOS DE USO.....	31
2.1.9.- ESPECIFICACIÓN DE LOS CASOS DE USO DEL SISTEMA	33
2.2.- ANÁLISIS	46
2.2.1.- MODELO DE ANÁLISIS	47
2.2.1.2.- Diagramas de Clase.....	47
2.2.1.3.- Diagramas de Colaboración	47
2.2.1.4.- Identificación de clases, responsabilidades y atributos	77
2.2.1.5.- Diagrama de Paquetes de la Aplicación	91
2.2.1.6.- Dependencia entre Paquetes de la Aplicación	94
2.3.- COMPARACIÓN DE TECNOLOGÍAS PARA EL USO DE MENSAJERA ELECTRÓNICA.....	95
2.4.- DISEÑO	101
2.4.1.- MODELO DE DISEÑO.....	101
2.4.2.- DIAGRAMAS DE SECUENCIA.....	101
2.4.3.- ESPECIFICACIÓN DE PAQUETES DE DISEÑO PARA LA APLICACIÓN	124

3.- IMPLEMENTACIÓN Y PRUEBAS DEL SISTEMA.....	128
3.1.- IMPLEMENTACIÓN.....	128
3.1.1.- MODELO DE IMPLEMENTACIÓN.....	128
3.1.2.- MODELO DE IMPLEMENTACIÓN PARA LA APLICACIÓN	128
3.1.3.- DESCRIPCIÓN DE CLASES POR COMPONENTES Y/O SUBSISTEMAS	129
3.1.4.- AMBIENTE DE DESARROLLO	131
3.1.5.- PRUEBAS DEL SISTEMA.....	132
3.1.6.- MODELO DE PRUEBAS	132
4.- CASO DE ESTUDIO.....	133
4.1.- DEFINICIÓN DEL CASO DE ESTUDIO	133
4.2.- INSTALACIÓN DEL CASO DE ESTUDIO.....	134
4.3.- INGRESO DE INFORMACIÓN AL SISTEMA	135
4.3.1.- CARGA DE DATOS.....	135
4.3.2.- EVALUACIÓN	135
4.4.- ANÁLISIS DE RESULTADOS	137
5.- CONCLUSIONES Y RECOMENDACIONES	139
5.1.- CONCLUSIONES	139
5.2.- RECOMENDACIONES.....	141
BIBLIOGRAFÍA.....	142
ANEXOS	144

RESUMEN

El presente proyecto se enfoca en el desarrollo de un sistema de colaboración el cual tiene como objetivo implantar funciones que sirvan de apoyo para la gestión comunicacional dentro de una organización.

En primer lugar se muestra un enfoque sobre la comunicación en una organización, y cuál es el objetivo de un sistema de colaboración.

En segundo lugar se muestra las herramientas de desarrollo de código abierto sus potencialidades y las licencias que se pueden utilizar.

En tercer lugar se muestra las herramientas de mensajería electrónica, su proyección a futuro, el impacto de uso y beneficios que tiene la comunicación para ambientes públicos y para organizaciones.

En cuarto lugar se analiza el proceso de desarrollo del sistema y una descripción del modelo de desarrollo que interviene en este proyecto.

En quinto lugar se procede con las fases y flujos de trabajo en el cual se analiza, diseña e implementa los componentes y la arquitectura del sistema.

En sexto lugar se procede con la implantación del sistema y las pruebas sobre el mismo.

En séptimo lugar se muestra un ejemplo con un caso de estudio para ver la factibilidad de éxito del sistema en un ambiente real.

En último lugar se adjunta las conclusiones y recomendaciones sobre el desarrollo del presente proyecto.

1.- MARCO TEÓRICO

1.1.- SISTEMAS DE COLABORACIÓN

1.1.1.- INTRODUCCIÓN

La tecnología marca la pauta para un nuevo modelo de productividad, este modelo toma como base los objetivos de la Empresa, su visión y misión. Por lo cual una empresa utiliza su estructura organizativa como sistema de productividad de la misma.

El avance tecnológico y el ingreso al mercado del comercio electrónico han cambiado el modo de trabajar de las personas en las empresas, lo cual ha creado nuevos estándares de trabajo a base de sistemas de software, sistemas de información que permitan agilizar los procesos.

Básicamente las tecnologías de cómputo se han enfocado en incrementar la productividad, con lo cual se ha descuidado la administración de documentos y la comunicación entre los diferentes actores que manejan los procesos en la empresa, siendo esta la base de los procesos del negocio, y que simplemente no existían, por lo cual se necesitaba de una herramienta que lo solucionara.

Uno de los sistemas que lo soluciona es el GroupWare¹ o Sistemas De Colaboración y tiene como significado la colaboración y cooperación del personal de la organización para la formación de grupos de trabajo, los cuales ingresen en una red organizacional de información y colaboración.

Con los sistemas de GroupWare los grupos de trabajo pueden colaborar de una manera más inteligente, más rápida, y más productiva con lo cual logran hacer más con menos recursos.

¹ [13] MEIRA Carlos María, PARÓN Angeles, ROSANO Déborah, Las tecnologías de colaboración serán decisivas para el desarrollo de la industria en los próximos tres años.

Los sistemas de colaboración nacen con el fin de administrar el trabajo en grupo y solventar las deficiencias de la falta de administración en la información.

Los sistemas de colaboración deben ayudar a empresas a descubrir nuevas capacidades y competencias a través de la cooperación y del conocimiento con lo cual se puede añadir valor agregado a los procesos del negocio con el uso de tecnologías de la información y la comunicación.

1.1.2.- DEFINICIONES

Para el entendimiento del sistema de colaboración hay que definir conceptos fundamentales respondiendo a preguntas que tienen que ver con eso: ¿Qué es conocimiento? ¿Qué es colaboración? ¿Qué es GroupWare?, cada una de estas preguntas están relacionadas, esta relación conforma al sistema de colaboración.

1.1.2.1.- Conocimiento

Al conocimiento se lo define como: “Entender, advertir, saber” del significado de las cosas, asuntos, acciones, hechos, etc. Y que en nuestro caso es el acercamiento y entendimiento de la información base del proceso hacia los actores del proceso.

1.1.2.2.- Colaboración

La colaboración busca reunir ayuda para el logro de una meta o fin, el cual comparte las responsabilidades, conocimiento, esfuerzos, apoyo, tecnología y riesgos que una actividad pueda tener.

La colaboración busca superar actividades individuales con lo cual se muestra la eficiencia y eficacia de la realización de actividades en cooperación transmitiendo el conocimiento.

1.1.2.3.- Groupware

El GroupWare es un software que permite en las organizaciones comunicar, colaborar y coordinar procesos claves del negocio. Forma una memoria empresarial que permite compartir el conocimiento y experiencia en beneficio de la organización.

El GroupWare o Sistemas de Colaboración se basan en la unión de tecnologías claves que antes estaban separadas, tales como: la mensajería, la conferencia y los flujos de información dentro de una organización o entre diferentes organizaciones.

Los sistemas de colaboración proporcionan herramientas para que la comunicación, tanto interna como externa sea fundamental, la colaboración en los grupos de trabajo con el uso de espacios de trabajo virtuales y la coordinación de los procesos estratégicos, rediseñando la estructura del proceso de negocios para comunicar y crear mecanismos de colaboración así como implementar políticas bien definidas en la empresa. Véase referencia².

1.1.3.- CARACTERÍSTICAS

Para encontrar las características debemos responder a la siguiente pregunta.

¿Cómo aplicar el GroupWare?

1.1.3.1.- Aplicación del Groupware

Para aplicar el GroupWare hay que explotar las capacidades del personal de la empresa, estas capacidades pueden ser clasificadas en:

- Las habilidades
- El conocimiento potencial
- La experiencia

Esta clasificación puede ser discutida, ya que solo persigue ofrecer una visión global del ser humano. La diferencia entre capacidades adquiridas e innatas puede ser ampliada según la actividad de cada uno.

En las empresas, que tanto necesitan del trabajo de los seres humanos para vivir y desarrollarse, se da mucha importancia a este conjunto de capacidades, que se denomina potencial. El potencial es un estado latente del ser humano que sólo es puesto en actividad cuando tenemos que lograr algo.

El GroupWare tiene que explotar el potencial del personal para mejorar la productividad.

² [15] , NICOL Chris, LA ASOCIACIÓN PARA EL PROGRESO DE LAS COMUNICACIONES

Los grupos de trabajo tienen su potencial en la comunicación efectiva, en la coordinación de las actividades con lo cual se puede llegar a cumplir con las metas del negocio.

1.2.- HERRAMIENTAS OPEN SOURCE

1.2.1.- INTRODUCCIÓN

La orientación sobre el desarrollo con Open Source es la definición de aplicaciones o plataformas tecnológicas que se basan en licencias capaces de distribuir copias del software gratis o de pago, con la disponibilidad de manipular y modificar el código fuente, y con la posibilidad de volverlo a distribuir.

Hay que dejar en claro la definición de Open Source para que no se malentienda.

La frase “Open Source”³ se la debe definir como código abierto, lo cual significa que el software desarrollado con esta tecnología tiene la libertad de que su código fuente puede ser explorado como modificado pero no se refiere al precio del producto.

Existen sistemas desarrollados con este modelo que son de carácter gratuito y otros que tienen costo. Las licencias con las que estas se distribuyen son las que ponen la pauta para poder poner o no precio al producto.

La idea fundamental en la utilización de la tecnología Open Source es que cualquier persona, organización o empresa pueda participar en el desarrollo del proyecto⁴.

Todo esto significa que la tecnología Open Source es un modelo colaborativo, el cual no monopoliza el desarrollo del mismo y que sobre todo abre la posibilidad a que una o varias personas o empresa(s) participen, con lo cual se asegura una mayor cantidad de transferencia de conocimiento entre las personas, y las empresas que se dediquen a desarrollar sobre este modelo.

³ [9] FreeSoftware Copyright (C), “Categorías de Software Libre y No Libre”

⁴ [11] “FREE SOFTWARE FOUNDATION”, GNU General Public License

Entre las empresas que han optado como modelo de desarrollo al Open Source se destaca a: SUN, IBM, NEC, HP, COMPAQ, INTEL, RED HAT, NOVELL.

1.2.2.- VENTAJAS DEL OPEN SOURCE

El modelo Open Source se basa en los siguientes temas:

- Administración
- Seguridad
- Costos

1.2.2.1.- Administración

Al poder disponer del código fuente del producto permite que se lo pueda modificar y personalizar de acuerdo a las necesidades, esto evita que se tenga que depender de proveedores con lo cual la administración del producto es más fácil.

También tenemos la posibilidad de realizar actualizaciones de acuerdo a nuestras necesidades ya que podemos elegir a diferentes proveedores con lo cual las alternativas del producto son varias.

1.2.2.2.- Seguridad

Este tema es de vital importancia ya que el producto debe tener bien clara y establecida la protección de la información.

Las empresas tienen en su información la base de los negocios, entonces el producto al poder administrar el código fuente, puede personalizarlo con las respectivas seguridades que se necesiten así como poder alterar, modificar posibles problemas o errores que tenga el sistema.

1.2.2.3.- Costos

Para tener clara la idea del manejo de costos tenemos que conocer algunas de las principales licencias con que se trabaja al utilizar un modelo de desarrollo Open Source.

Cabe recalcar que el Software Libre no significa gratis, el producto dependerá de la licencia que cada persona, grupo o empresa desarrolladora defina y con esto saber si este tendrá un costo o será de libre distribución.

El estudio de las licencias que se aplican al producto permitirá su modificación, reutilización de código, actualización, etc.

Entre los tipos de licencias⁵ existentes tenemos:

GPL “General Public License”.- Creada por la Free Software Foundation, la cual permite la distribución de copias del software, gratis o de pago, modificar el software recibido y volverlo a distribuir, de la exploración del código fuente, etc. Esta licencia también afecta al cliente que recibe esta licencia con la cual esta prohibido que se nieguen estos mismos derechos.

LGPL “Lesser General Public License”. - Creada por la Free Software Foundation, esta es una variación de la GPL y se la creo para casos especiales, como por ejemplo para evitar la utilización de librerías por un programa comercial.

SPL “Sun Public License”.- Creada por la Sun con el objetivo de poner términos y normas para el uso de sus productos.

CDDL “Common Development and Distribution License”. - Es una licencia abierta de la fuente que toma del explorador Mozilla, y cuya licencia (MPL) es de carácter pública, lo hace reutilizable sin permiso de modificación. Es una licencia que tiene derecho de copia y que proporciona las protecciones y libertades necesarias para la fuente.

1.2.3.- REGULACIÓN DE LAS LICENCIAS

Existe una entidad llamada OSI “Open Source Initiative” la cual es la que regula y aprueba las diferentes licencias para el software libre.

⁵ [9] FreeSoftware Copyright (C), “Categorías de Software Libre y No Libre”

1.2.4.- LOS RÉDITOS DEL OPEN SOURCE

Cuando se desarrolla con el modelo de software libre la mayoría del software creado tiene la licencia GPL sin costo, pero si tiene costo la adaptación, personalización y soporte del software al cliente como se muestra en la Figura 1.1 pero mucho menor a tener un producto de software propietario. Véase referencia⁶.



[FIGURA 1.1]

Realizado por: El Autor

Referencia: El Autor

Además de los réditos económicos se encuentran los réditos del conocimiento ya que uno de los principales objetivos del Open Source es el de permitir la transferencia de conocimientos por parte de una persona o grupo sobre la forma de desarrollar, los métodos, las ideas y ayudas claves para mejorar el desarrollo así como el de que otras personas o grupos colaboren con el desarrollo y mejoramiento de un producto de software. Véase referencia⁷.

1.3.- SISTEMAS DE MENSAJERÍA

1.3.1.- CONCEPTOS

⁶ [9] GNU General Public License, FREE SOFTWARE FUNDATION

⁷ [14] MOLPECERES Alberto, PEREZ Martín. Arquitectura Empresarial y Software Libre, J2EE, Segunda Revisión

Mensaje.- Es un medio que permite la comunicación y colaboración entre personas y que puede ser enviada y recibida por medios: físicos, visuales, auditivos, etc.

El sistema de mensajería busca proveer una solución a la falta de comunicación con el uso de Internet a través de la mensajería electrónica.

Mensajería Electrónica.- Con el uso del Internet existen varios sistemas de mensajería electrónica que permiten el envío y recepción de mensajes.

Correo Electrónico.- Este sistema envía mensajes electrónicos usualmente utilizando los protocolos: SMTP, POP3, IMAP. Todos estos protocolos funcionan como agentes de transferencia de correo y otros como agentes usuarios de correo.

Network News Transport Protocol (NNTP).- Es un protocolo de transferencia de noticias. Es el Protocolo de red utilizado por el Usenet Internet service. Es un Protocolo de red basado en tiras de textos enviados sobre canales TCP de 7 BIT ASCII. Es usado para subir y bajar así como para transferir artículos entre servidores.

Chat.- Es un sistema que permite que varios usuarios ingresen a un programa servidor, que permite el envío de mensajes y que todos los usuarios puedan verlo, y que cualquiera responda, en los Chat's no es común que presten mayores servicios que el de transferir mensajes a un determinado grupo.

La mensajería instantánea es un punto intermedio entre el correo electrónico y los sistemas de Chat, ya que la mensajería depende directamente del Internet, además ofrecen una comunicación en línea donde se escribe el mensaje en una ventana tipo texto plano y que se lo envía a uno o varios destinatarios.

Sabemos que una buena comunicación organizacional crea un buen ambiente laboral y que la colaboración y comunicación son fundamentales dentro de una organización. A mayor comunicación mayor es el grado de ahorro de dinero, con

lo cual muchas organizaciones han optado por el uso de sistemas públicos como son: el MSN Messenger, el ICQ y el Yahoo, como sistemas de mensajería. Véase referencias ⁸ ⁹.

1.3.2.- ESTADÍSTICAS DE USO DE LOS SISTEMAS DE MENSAJERÍA INSTANTÁNEA

Para conocer el impacto que tendrá la implantación de un sistema interno de mensajería en una organización se obtuvo el análisis publicado por la empresa consultora “Radicati Group”¹⁰ en la cual se expresa lo siguiente:

Cabe citar que todo el informe descrito sobre el impacto de la mensajería instantánea está publicado por la empresa Radicati en la dirección web que consta en la referencia y es de libre acceso a esta información.

El grupo Radicati publicó un análisis sobre el mercado de la mensajería instantánea para el período 2005 a 2009, en el cual informa lo siguiente:

- El 85% de todas las empresas en Estados Unidos utilizan la mensajería instantánea, bien sea para uso personal o para uso del negocio.
- El 20% de todas las empresas a nivel global usan la mensajería instantánea y se espera que esta cifra ascienda al 80% para el 2008
- Al final del 2008 habrá 72 millones de usuarios corporativos de mensajería instantánea
- Al finalizar el 2008, 88% de los usuarios corporativos de mensajería instantánea dependerán de una red pública de mensajería instantánea.

En el mismo estudio se informó que las razones citadas por las empresas para desplegar soluciones de mensajería instantánea fueron:

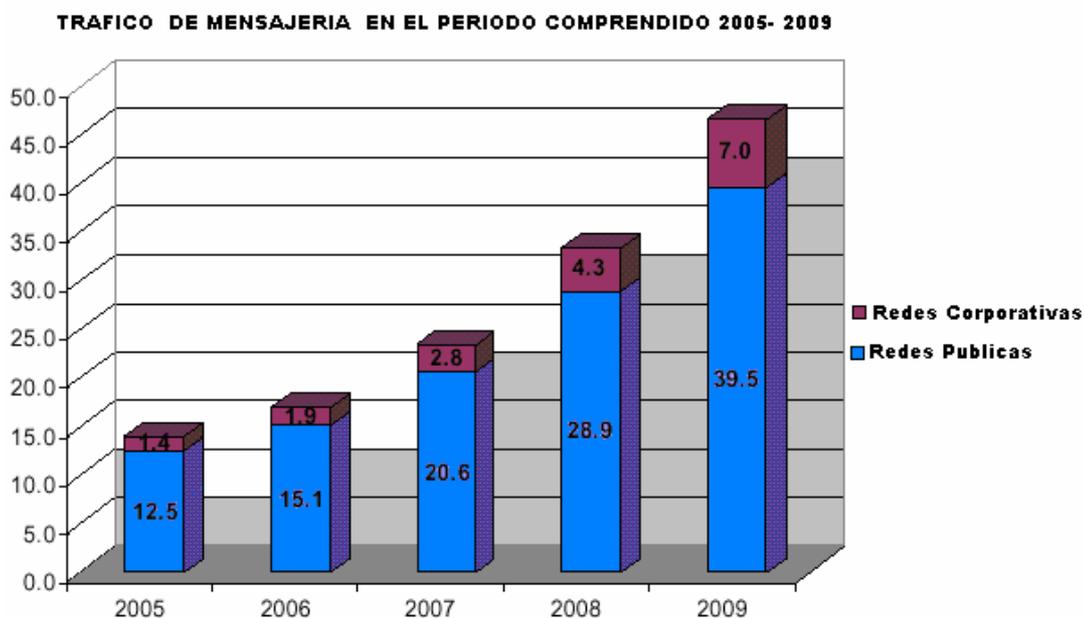
- 44% Aumentar las comunicaciones Inter-oficinas.

⁸ [6] ARMELINI Guillermo, Nuevos retos para la mensajería instantánea

⁹ [15] NICOL Chris, LA ASOCIACIÓN PARA EL PROGRESO DE LAS COMUNICACIONES

¹⁰ El Grupo Radicati, S.A., URL: <http://www.radicati.com/informemensajeria.aspx>, 2005

- 33% Reducir costos de llamadas telefónicas de larga distancia
- 11% Aumentar productividad
- 11% Complementar sistemas actuales de correo y teléfono



[FIGURA 1.2]

Realizado por: El autor

Referencia: <http://www.radicati.com/informemensajeria.aspx>

Como se muestra en la Figura 1.2 los indicadores proyectan un crecimiento del uso de mecanismos de mensajería en lugares públicos como en organizaciones o empresas.

Este informe fue desarrollado para Clientes Corporativos, para los Proveedores de servicios de Internet, para los Vendedores, y para los Inversionistas que necesitan tomar decisiones de la proyección de la mensajería instantánea y su uso en redes públicas como en redes corporativas.

1.4.- METODOLOGÍA DE DESARROLLO

1.4.1.- INTRODUCCIÓN

Para hacer uso de una metodología acorde al proyecto hay que tomar en cuenta ciertos aspectos que tiene el proceso para el desarrollo del sistema de software.

1.4.2.- EL CICLO DE VIDA

El ciclo de vida del software define los procesos que el producto de software va a tener, siendo esta la base principal de la metodología ya que complementa con sus características y objetivos al cumplimiento de las necesidades que se tiene para el desarrollo productivo del software.

1.4.3.- LOS PROCESOS DEL CICLO DE VIDA DEL SOFTWARE

El ciclo de vida del software básicamente se clasifica en tres procesos fundamentales:

- Procesos principales
- Procesos de soporte
- Procesos de la organización

Los procesos principales.- Son los procesos útiles a las personas que realizan operaciones de uso, mantenimiento, desarrollo, durante el ciclo de vida del software.

Los procesos de soporte.- Son procesos que sirven como apoyo a cualquiera de los procesos en cualquier punto del ciclo de vida del software.

Los procesos de la organización.- Son procesos que utilizan organizaciones con el fin de cumplir funciones tales como la gestión, organización, capacitación del personal para mejorar los procesos y así conseguir una organización mas efectiva. Véase referencia¹¹.

1.4.4.- PROCESOS DEL DESARROLLO DE SOFTWARE

Estos son los procesos base para el desarrollo del software:

- Análisis y Requisitos del Sistema.
- Diseño de la arquitectura del Sistema

¹¹ [10] FERNÁNDEZ SÁNCHEZ José Luís, Reusabilidad y Desarrollo Orientado a Objetos

- Análisis de los requisitos del Software
- Diseño de la arquitectura del Software
- Diseño detallado del Software
- Codificación y pruebas del Software
- Integración del Software
- Prueba del Software
- Integración del Sistema
- Prueba del Sistema
- Instalación del Software
- Soporte del proceso de aceptación del Software

Todos los procesos descritos forman parte del desarrollo del software, con lo cual hace falta un orden lógico y que se acople a las necesidades de desarrollo.

Los Modelos de desarrollo cumplen estas funciones, cada modelo se caracteriza por realizar sus respectivas actividades, también describen su manera de manejar los procesos y proporcionar una guía para el desarrollo del proyecto.

Teniendo en cuenta el ciclo de vida del software, los procesos a desarrollarse, la clasificación de los procesos, y la complejidad de este desarrollo, se debe optar por una metodología que cumplan estos retos. Véase referencia¹².

1.4.5.- PROCESO UNIFICADO DE DESARROLLO DEL SOFTWARE

Esta metodología de desarrollo se basa en un modelo en espiral capaz de organizar las iteraciones por etapas y fases para obtener una estructura más sólida, clara y ajustable a las necesidades del desarrollo del proyecto.

Este modelo tiene como objetivo la viabilidad del proyecto y la definición de planes de manejo para garantizar desde las fases iniciales la eliminación o mitigación de riesgos en donde se permite hacer un proceso continuo de pruebas y retroalimentación.

Las ventajas de utilizar el modelo en espiral son:

¹² [12] MENDOZA Sánchez María, Metodologías De Desarrollo De Software

- Desarrollar un producto con calidad es el objetivo primario.
- Se basa en la reutilización de componentes.
- Elimina posibles errores descubiertos en las fases iniciales.
- Es iterativo e incremental.
- Integra desarrollo con mantenimiento.
- Provee un marco de desarrollo de hardware y software.

1.4.5.1.- Historia

La historia del Proceso Unificado es una introducción para conocer la evolución del proceso y con lo cual se hace referencia a las personas que participaron en la creación del mismo.

El Proceso Unificado de desarrollo nace de la unión de metodologías de diferentes autores, los cuales formaron un método claro, ordenado, robusto y capaz de proporcionar normas para el desarrollo de sistemas de software.

Nace con la empresa Ericsson la cual buscaba crear sistemas para sus productos basados en componentes, los mismos que utilizaban bloques los cuales mostraban las asociaciones para formar la comunicación. Los bloques significaban los factores críticos del negocio y de tal manera que separaban los componentes. Con esta forma de desarrollo lo más importante eran los componentes.

El creador de este estándar trabajó para Ericsson, Ivar Jacobson, quien se separó de Ericsson en el año 1987 con el fin de fundar su propia empresa la cual se llamo Objectory. Su trabajo radicó en la creación de un proceso denominado OBJECTORY o fábrica de Objetos, con lo cual se clarificaba la idea de la del manejo de Objetos.

La orientación a objetos nace como consecuencia de los cambios tecnológicos de los sistemas informáticos, todo esto basándose en formas de codificación nuevas de menor costo y mayor potencia para el análisis y diseño de nuevos productos de software que sean más estables, ya que los anteriores lenguajes de programación se hacían muy complejos y sobre todo volátiles.

Los lenguajes orientados a Objetos permiten más simplicidad para manejar sistemas orientados a usuarios.

Después de que Ivar Jacobson creara objectory a finales del año 1995 pasa a unificarse con la empresa Rational Software Corporation para unificar procesos de desarrollo.

Rational se decide a crear un entorno que mejore la productividad del desarrollo de sistemas de software y que para esos momentos la metodología del desarrollo de software era fundamental.

Rational continuó en su proceso de producir una herramienta sólida, y empiezo con el desarrollo iterativo con lo cual trato de crear mini proyectos. Rational para esto contó con más colaboradores, los cuales estuvieron con esta empresa desde sus inicios y entre los cuales destacamos a Grandy Booch el cual con su aporte brindó conceptos fundamentales sobre la arquitectura y la iteración.

“Un estilo de desarrollo dirigido por la arquitectura es normalmente la mejor aproximación para la creación de la mayoría de los proyectos complejos basados en el software.”

“Para que un proyecto orientado a objetos tenga éxito, debe aplicarse un proceso iterativo e incremental.”

Rational continuó con el desarrollo de un modelo de desarrollo robusto, con lo cual combinó el proceso Objectory, la experiencia y la práctica para formar un nuevo proceso, el proceso objectory de Rational.

El proceso Objectory de Rational se incorporó al desarrollo iterativo modificando características y que ahora evalúa los riesgos del desarrollo en base a la arquitectura.

En este proceso Rational necesitó de un lenguaje capaz de acoplarse a las necesidades del proceso Objectory de Rational,

UML es el lenguaje desarrollado y escogido por Rational el cual va a aportar un modelado visual y consistente del proceso de desarrollo. UML Expresa los resultados de las metodologías Orientadas a Objetos.

En este proceso también se incorporó James Rumbaugh quien aporta con métodos de modelaje de objetos.

Con estos tres componentes se publicó una nueva versión del método unificado de desarrollo.

Casi al mismo tiempo Jacobson se incorporó al grupo de desarrolladores del lenguaje UML con lo cual también sacaron una nueva versión del lenguaje de modelamiento unificado "UML".

Para el año de 1998 la empresa Rational tenía un proceso de desarrollo claro, organizado y dinámico.

Entonces gracias al aporte y colaboración de diferentes técnicas de desarrollo, a través del uso del lenguaje de modelamiento unificado, a las varias empresas que colaboraron con aportes al desarrollo del modelo, la empresa Rational estandariza el modelo de desarrollo como el "Proceso Unificado de desarrollo de Software". Véase referencia¹³.

1.4.5.2.- Características del Proceso Unificado de Desarrollo de Software

El proceso unificado de desarrollo se basa en tres pilares fundamentales que son:

- Dirigido por casos de uso
- Centrado en la arquitectura
- Iterativo e Incremental

1.4.5.3.- Casos de Uso

El caso de uso se lo define como la representación de una interacción que sucede para que se cumpla un determinado proceso, entre el usuario y el proceso.

El caso de uso debe cumplir con las expectativas y necesidades del usuario.

Esta representación por caso de uso se maneja por la necesidad de buscar y representar los verdaderos requisitos que para el desarrollo del proyecto de software se debe de encontrar.

Los casos de uso explican la secuencia de acciones de una manera fácil y concisa de tal manera que al momento de desarrollar el sistema sea el elemento director del proceso de desarrollo. Véase referencia¹⁴.

Utilidad de los Casos de Uso:

- Es una manera de capturar los requisitos del proyecto.

¹³ [3] JACOBSON Ivar, BOOCH Grady, RUMBAUGH James, El Proceso Unificado de Desarrollo de Software

¹⁴ [16] SÁNCHEZ PÉREZ Javier, Metodologías de Desarrollo Software - Proceso Unificado de Desarrollo de Software

- Al diseñar un modelo de casos de uso se le brinda un valor añadido al usuario.
- Los casos de uso determinan el orden de implementación e integración de los componentes.
- Los casos de uso proporcionan una clara visión y el complemento para el diseño de las demás fases del proceso unificado de desarrollo.
- Los casos de uso enlazan los flujos de trabajo del proceso unificado de desarrollo.

1.4.5.4.- Modelos de Casos y Usos

Los modelos de casos y usos son básicamente el conjunto de casos de uso que describe la funcionalidad del sistema en su totalidad.

El proceso unificado de desarrollo utiliza estos modelos no solo para la especificación de requerimientos del sistema sino también para el diseño, implementación y pruebas del mismo.

Los modelos requeridos en el proceso unificado de desarrollo son:

- MODELO DE CASOS DE USO
- MODELO DE ANÁLISIS
- MODELOS DEL DISEÑO
- MODELO DE DESPLIEGUE
- MODELOS DE CASOS DE PRUEBA

Al modelo se lo considera como una abstracción del sistema o de una parte del sistema, el cual se encuentra modelado desde el punto de vista del modelador.

Como existe un flujo de trabajo en el proceso unificado, este tiende a tener arreglos o modificaciones los cuales son dependientes del modelo que lo generó.

Por lo cual, el flujo de trabajo tiende a unir a los modelos a través de trazas, las cuales permitirán realizar las correcciones desde el modelo base.

1.4.5.5.- El ciclo de vida del Proceso Unificado de Desarrollo

Se repite a lo largo de una serie de ciclos siendo los ciclos la vida del sistema. Cada ciclo concluye con una versión del software.

Cada ciclo consta de cuatro fases: Inicio, Elaboración, Construcción y Transición.

Fase de Inicio.- Es una descripción del producto final a partir de una buena idea, se presenta un modelo de casos de uso simplificado que contenga los casos de uso más críticos.

Fase de Elaboración.- Se especifica en detalle los casos de uso del producto y se diseña la arquitectura del sistema.

Fase de Construcción.- Se crea el producto, aparecen las versiones en base a la dirección que tuvo el mismo.

Fase de Transición.- Fase que se dedica a la corrección de errores, mejoras, y proporciona asistencia y ayuda para el desarrollo y manejo del producto. Véase referencia¹⁵.

1.4.5.6.- Iterativo e Incremental

Cada fase se divide en iteraciones.

La iteración es un paso del flujo de trabajo para el desarrollo del producto, la iteración produce los incrementos con lo cual el producto empieza a crecer.

En cada iteración, los desarrolladores identifican y especifican los casos de uso relevantes, utilizando la arquitectura seleccionada como guía.

Implementan el diseño utilizando componentes y verifican que los componentes satisfagan los casos de uso.

Las iteraciones deben pasar por los cinco flujos de trabajo.

- Requisitos
- Análisis
- Diseño
- Implementación
- Pruebas.

¹⁵ [3] JACOBSON Ivar, BOOCH Grady, RUMBAUGH James, El Proceso Unificado de Desarrollo de Software

REQUISITOS

En el flujo de requisitos se recoge la información necesaria para comenzar el proceso de desarrollo del proyecto, es la base con la cual se desarrollan los flujos subsiguientes. La captura de requisitos se basa en el modelo de casos de uso el cual incluye los casos de uso del sistema en los que se muestran los roles o funcionalidades del mismo y los actores que hacen uso del caso de uso, a este conjunto de interacciones entre actores y casos de uso se lo conoce como modelos de casos de uso del sistema. La captura de requisitos permite recopilar información necesaria para desarrollar el sistema y esta se la consigue en concordancia con los clientes y los capturadores de requisitos, el modelo de casos de uso se convertirá en el acuerdo que permita avanzar en el proceso de desarrollo. El modelo de casos de uso contiene artefactos que muestran la interacción del sistema con los usuarios, en el sistema encontramos funcionalidades o casos de uso, los actores o usuarios, las relaciones entre los dos, las generalidades, los diagramas de estado. El caso de uso debe cumplir con la funcionalidad para la cual fue creada y debe cumplir su atomicidad en el momento de su ejecución. En la captura de requisitos se define el modelo de casos de uso, la descripción de cada caso de uso, un prototipo de las interfaces, las relaciones de los actores con los casos de uso y el glosario de términos asociados al modelo de casos de uso.

ANÁLISIS

El análisis proporciona una visión general del sistema que puede ser más difícil de obtener mediante el estudio de los resultados del diseño y la implementación. En el flujo de análisis se procede a analizar los requisitos del sistema con un nuevo artefacto llamado modelo de análisis, en el cual los requisitos del sistema son refinados y estructurados de tal forma que son comprensibles para los modeladores del sistema. El modelo de análisis se basa en artefactos llamados clases de análisis, las cuales a su vez se encuentran divididas en estereotipos de interfaz, de control y de entidad. Las clases de análisis representan las abstracciones de una o varias clases del diseño del sistema denotando el

contexto del dominio del problema de manera más conceptual. Las clases definen atributos los cuales definen conceptos que son única y exclusivamente reconocibles en el dominio del problema.

El modelo de análisis ofrece como respuesta:

- Especificación más precisa de los requisitos del proyecto.
- Descripción del funcionamiento interno del sistema.
- Estructura los requisitos facilitando su comprensión.
- El modelo de análisis tiene la función de preparar el camino al modelo de diseño siendo cada uno un modelo por si mismo.

DISEÑO

En el flujo de diseño del sistema se modela el sistema proporcionando la forma que de soporte a los requerimientos funcionales como no funcionales, para esto se basa y se toma como entrada al modelo de análisis el cual proporciona aspectos más detallados de las necesidades del sistema.

El diseño tiene como objetivo el comprender los requisitos no funcionales y restricciones técnicas, y es la base para continuar con la fase de implementación ya que la fase de diseño proporciona una separación del trabajo que se debe realizar en la fase de implementación.

También contribuye a una arquitectura estable por ser el soporte de la fase de implementación.

IMPLEMENTACIÓN

En la fase de implementación se procede a crear ficheros en los cuales se pondrá el código fuente, scripts, ejecutables, etc., lo cual es el resultado del proceso siguiente de la fase de análisis.

El modelo de implementación denota la implementación del sistema en términos de componentes los cuales son distribuidos a los nodos en el diagrama de despliegue.

En la fase de implementación también se implementa las clases y subsistemas del sistema que fueron encontradas en la fase de diseño.

El artefacto que denota la fase de implementación es el modelo de implementación el cual se compone de:

- Componentes
- Subsistemas
- Interfaces
- Arquitectura del modelo de implementación

Al tener los artefactos descritos anteriormente se procede a construir un plan de integración de construcciones el cual denotará la funcionalidad ya en construcción del sistema.

PRUEBAS

En esta fase se procede a verificar el resultado de la implementación con la entrega de una versión del sistema que es sometida a pruebas de una manera planificada.

El artefacto utilizado en la fase de pruebas es el modelo de pruebas que esta constituido por los siguientes artefactos:

- Casos de prueba
- Plan de pruebas
- Procedimientos
- Defectos
- Evaluación del pruebas

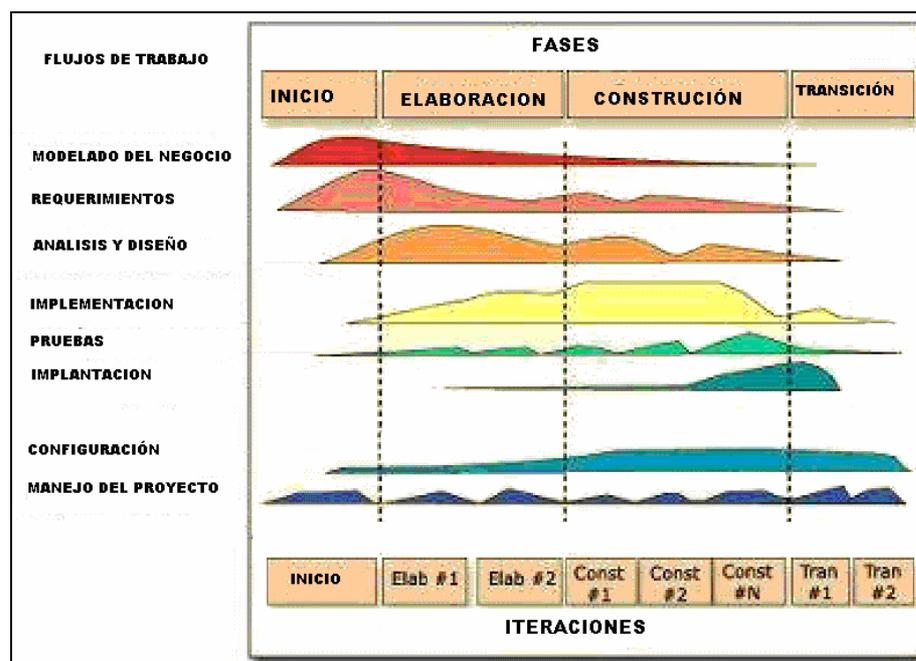
Los casos de prueba deben contener datos precisos que brinde información para la realización de la evaluación con lo cual se procede a definir el grado de aprobación del sistema y su respectiva versión, ya sea una versión candidata o versión final. Véase referencia¹⁶.

¹⁶ [16] SÁNCHEZ PÉREZ Javier, Metodologías de Desarrollo Software - Proceso Unificado de Desarrollo de Software

1.4.5.7.- Centrado en la Arquitectura

La arquitectura del sistema será el resultado del cumplimiento de fases de requisitos y de análisis con lo cual se obtendrá un modelo de diseño que permita el modelaje de una arquitectura capaz de adaptarse y soportar los diferentes requisitos del sistema.

El proceso unificado se centra en la arquitectura porque se realizaron pasos que identifiquen de forma adecuada los requisitos del sistema, analizándolos y definiendo los componentes que intervendrán en el sistema, de tal forma que el sistema tenga un soporte desde las bases, comenzando con la captura de los requisitos hasta la implementación de la versión final del sistema incluyendo su mantenimiento. En la Figura 1.3 se muestra como el flujo de trabajo se distribuye con respecto a las diferentes fases en el desarrollo del sistema.



[FIGURA 1.3]

Realizado por: El autor

Referencia: Metodologías de Desarrollo Software - Proceso Unificado de Desarrollo de Software

2.- ANÁLISIS Y DISEÑO DEL SISTEMA

2.1.- REQUERIMIENTOS

2.1.1.- PLANTEAMIENTO DEL PROBLEMA

La comunicación efectiva en una empresa provee de un acercamiento oportuno al momento de tomar decisiones que afecten en una negociación, teniendo en consideración que el objetivo de una empresa es el negocio.

Si la falta de comunicación afecta directamente para llegar a una negociación exitosa entonces debemos reestructurar los mecanismos que nos permitan mejorar u optimizar la manera de comunicarse en una empresa. Además de no aprovechar los recursos tecnológicos que tenemos hoy en día haciendo uso de los tradicionales sistemas de comunicación como son el teléfono, el fax, etc.

Por lo tanto es recomendable cambiar la situación actual a fin de mejorar y disminuir tiempo, disminuir costos, optimizar recursos y ganar en colaboración empresarial. De no ser así con el tiempo puede llegar la empresa a estancarse en punto que no le permita crecer y perder en el mercado en el que se trabaje.

2.1.2.- FORMULACIÓN DEL PROBLEMA

¿Qué se puede hacer en vista de la falta de una comunicación efectiva por parte de los socios, directivos, trabajadores de una empresa para la mejor toma de decisiones en pro del beneficio de la empresa y haciendo uso de soluciones tecnológicas?

2.1.3.- OBJETIVO GENERAL DEL PROYECTO

Implementar un sistema de colaboración en línea que interactúe en tiempo real con los usuarios, y con funcionalidades que le permitan a los socios, directivos y trabajadores de la empresa formar y ser parte de un ambiente de ayuda, diálogo y discusión, permitiéndole a la empresa crecer y no perder en el mercado en el que se desempeñe.

2.1.4.- REQUISITOS FUNCIONALES Y NO FUNCIONALES

FUNCIONALES

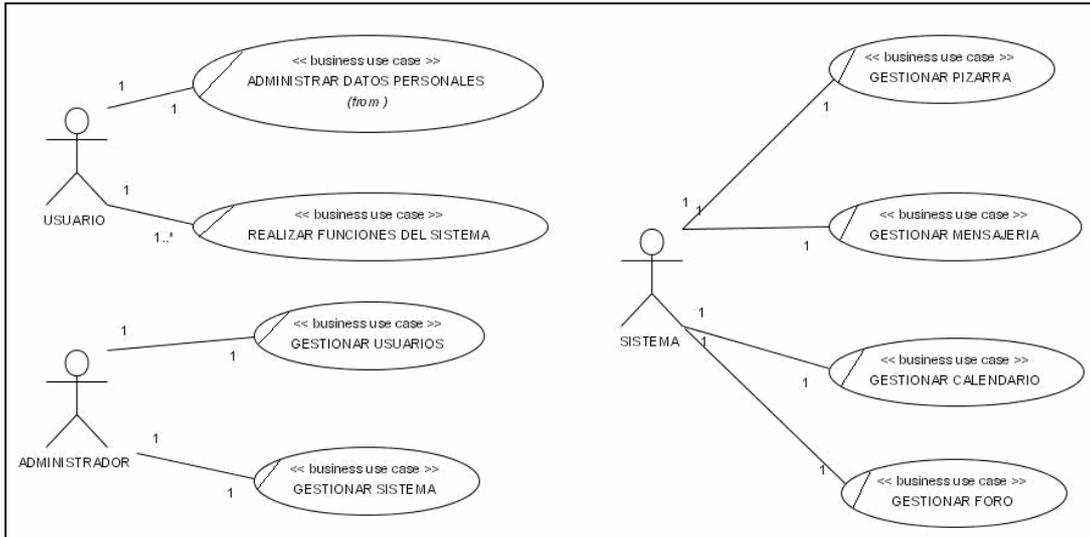
- Calendario compartido para citas y eventos.
- Libreta de tareas.
- Sistema de mensajería instantánea.
- Categorización de usuarios del sistema.
- Foros de dialogo y discusión.
- Pizarra virtual.

NO FUNCIONALES

- Lenguaje orientado a objetos.
- Arquitectura del sistemas distribuido en capas.
- Servidores de datos, de aplicaciones Web y de aplicaciones del negocio que sean herramientas de código abierto.
- Lenguaje de manejo de objetos Web.
- Un navegador que permita la ejecución de código en pantalla del cliente.

2.1.5.- MODELO DEL NEGOCIO

El modelo del negocio que se muestra en la Figura 2.1 representa una abstracción simplificada de los aspectos que comprenden el negocio y la cual está representada por un modelo de casos de uso, y en la cual se basa el sistema, en este caso se representa funcionalidades macro del sistema.



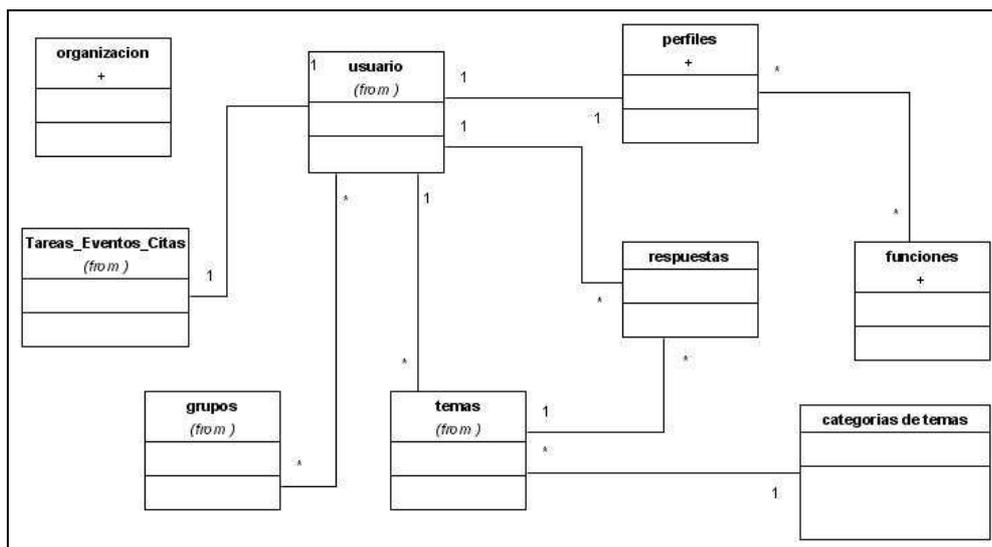
[Figura 2.1]

Realizado por: El autor

Referencia: Modelos desarrollados por el autor con la herramienta Poseidón.

2.1.6.- MODELO CONCEPTUAL O DEL DOMINIO

El modelo de dominio del sistema de colaboración que esta representado en la Figura 2.2 muestra las clases que se encuentran identificadas dentro del contexto del sistema con lo cual se comprende de mejor manera los requisitos del sistema así como los términos usados en el desarrollo del sistema de colaboración.



[Figura 2.2]

Realizado por: El autor

Referencia: Modelos desarrollados por el autor con la herramienta Poseidón.

2.1.7.- IDENTIFICACIÓN DE ACTORES Y CASOS DE USO

ACTORES DEL SISTEMA

USUARIO.- Es la persona que hará uso el sistema para participar en los foros de discusión, el uso del calendario, el manejo de tareas, la categorización de usuarios en grupos de trabajo, el uso de la pizarra virtual, y el sistema de mensajería instantánea.

ADMINISTRADOR.- Es un usuario con perfil de administrador que realizará tareas de gestión de usuarios, gestión de perfiles, gestión del sistema, gestión de funciones.

SISTEMA.- Será el encargado de actualización de la información de la pizarra virtual y del envío de mensajes.

2.1.7.1.- Casos de Uso del Sistema

NOMBRE	DESCRIPCIÓN
Gestionar Usuarios	El administrador utiliza este caso de uso para ingresar la información de un nuevo usuario que tenga un perfil con el cual pueda usar el sistema.
	El administrador utiliza este caso de uso para eliminar la información de un usuario del sistema.
	El administrador utiliza este caso de uso para consultar los usuarios registrados y existentes en el sistema.
	El administrador utiliza este caso de uso para modificar la información de un usuario del sistema así como su perfil.

Gestionar Perfiles	El administrador utiliza este caso de uso para ingresar un perfil el cual contenga la información de los derechos y privilegios de los usuarios en el sistema.
	El administrador utiliza este caso de uso para consultar los perfiles que hay en el sistema.
	El administrador utiliza este caso de uso para eliminar los perfiles que existen en el sistema.
	El administrador utiliza este caso de uso para modificar los perfiles existentes en el sistema.

Gestionar Organización	El administrador utiliza este caso de uso para ingresar la información de la organización o empresa en el sistema.
	El administrador utiliza este caso de uso para modificar la información ingresada de la organización.

Ingreso al sistema	Los usuarios o el administrador utilizan este caso de uso para registrar su ingreso al sistema.
--------------------	---

Envió de mensajes instantáneos	El usuario utiliza este caso de uso para empezar una conversación con otros usuarios que estén conectados en al sistema a través de mensajería instantánea.
--------------------------------	---

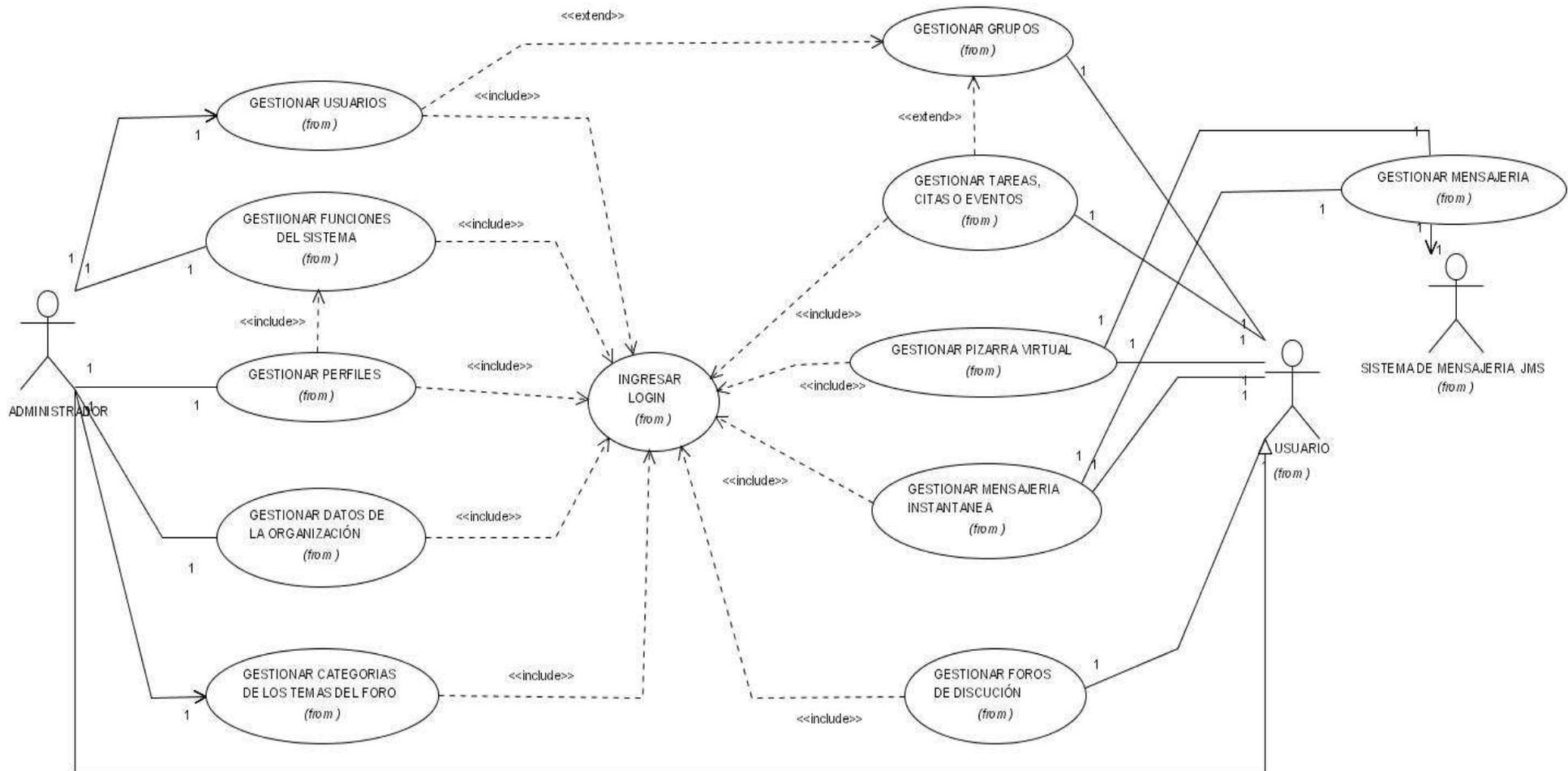
Gestionar Eventos o Citas	El usuario utiliza este caso de uso para consultar los eventos o citas que tiene y organizar sus actividades.
	El usuario utiliza este caso de uso para modificar los eventos o citas que ha ingresado en el sistema.
	El usuario utiliza este caso de uso para ingresar un evento o cita que tenga para llevar un registro y un cronograma de actividades.
	El usuario utiliza este caso de uso para eliminar un evento o cita que esta registrada en el sistema.

Ingresar a pizarra virtual	El usuario utiliza este caso de uso para iniciar una sesión de pizarra virtual con otros usuarios en línea.
----------------------------	---

Gestionar Grupos de usuarios	El usuario utiliza este caso de uso para organizar a más usuarios en grupos de trabajo con el fin de programar tareas, eventos, citas.
	El usuario utiliza este caso de uso para actualizar la información del grupo o la lista de integrantes.
	El usuario utiliza este caso de uso para eliminar grupos registrados en el sistema.
	El usuario utiliza este caso de uso para eliminar grupos registrados en el sistema.
	El usuario utiliza este caso de uso para consultar que grupos se encuentran registrados en el sistema.
Gestionar Temas de Foros de Discusión	El usuario utiliza este caso de uso para ingresar un tema para su discusión en el foro.
	El usuario utiliza este caso de uso para consultar que temas están disponibles en el foro para ser tratados por los demás integrantes.
	El usuario utiliza este caso de uso para modificar un tema registrado y que el haya ingresado.
	El usuario utiliza este caso de uso para eliminar un tema registrado y que el haga ingresado.
	El usuario utiliza este caso de uso para responder a un tema que previamente fue ingresado con el fin de dar su opinión.
Gestionar Funciones del sistema	El administrador utiliza este caso de uso para ingresar una función al sistema.
	El administrador utiliza este caso de uso para consultar las funciones que el sistema realiza.
	El administrador utiliza este caso de uso para eliminar las funciones que existen en el sistema.
	El administrador utiliza este caso de uso para modificar las funciones existentes en el sistema.

2.1.8.- MODELO DE CASOS DE USO

El modelo de casos de uso que se representa en la [Figura 2.3] muestra las interacciones del sistema con los usuarios así se describe parcialmente el sistema y nos muestra una visión mas clara de los requerimientos.



[Figura 2.3]

Realizado por: El autor

Referencia: Modelos desarrollados por el autor con la herramienta Poseidón.

2.1.9.- ESPECIFICACIÓN DE LOS CASOS DE USO DEL SISTEMA

Caso de Uso: Gestionar Usuarios
Objetivo: Realizar el ingreso, consulta, actualización y eliminación de usuarios que hagan uso del sistema.
Actores: Administrador(A), Sistema(S).
Precondiciones: Que el administrador este identificado y autenticado en el sistema. Que existan perfiles creados.
Postcondiciones:
<p>Pasos:</p> <p>Ingreso</p> <ol style="list-style-type: none"> 1. (A) Quiere crear un nuevo usuario. 2. (A) Ingresa los datos del usuario. 3. (S) Valida los datos del usuario. 4. (A) Escoge un perfil para el usuario. 5. (A) Confirma el ingreso aceptando. 6. (S) Registra al usuario y se asigna una identificación. 7. (S) Notifica el ingreso de un nuevo usuario listando los usuarios. <p>Consulta</p> <ol style="list-style-type: none"> 1. (A) Decide consultar los usuarios del sistema. 2. (S) Muestra lista o menú de opciones de búsqueda de usuarios. 3. (A) Escoge el usuario consultado. 4. (S) Muestra la información del usuario. <p>Actualización</p> <ol style="list-style-type: none"> 1. (A) Realiza pasos de Consulta. 2. (A) Edita o ingresa información del usuario. 3. (S) Valida los datos del usuario. 4. (A) Confirma la actualización aceptando. 5. (S) Registra los cambios en la información del usuario. 6. (S) Notifica la actualización del usuario listando los usuarios. <p>Eliminación</p> <ol style="list-style-type: none"> 1. (A) Realiza pasos de Consulta. 2. (S) Se confirma que el usuario no haya ingresado al sistema (Log in).

<p>3. (A) Confirma la eliminación aceptando.</p> <p>4. (S) Marca al usuario como eliminado.</p> <p>5. (S) Notifica la eliminación del usuario al administrador.</p>
<p>Extensiones:</p> <p>2.1-4.1-6.1-7.1 Se cancela cualquier acción.</p> <p>2.1.1-4.1.1-6.1.1-7.1.1 (S) Pide la confirmación de la cancelación.</p> <p>2.1.2-4.1.2-6.1.2-7.1.2 (A) Confirma la cancelación aceptando.</p> <p>2.1.3-4.1.3-6.1.3-7.1.3 (S) Finaliza el caso de uso.</p> <p>3.1 Los datos ingresados son incorrectos (Ingreso y Actualización).</p> <p>3.1.1 (S) Indica el error y regresa a flujo 2.</p> <p>3.1.2 (S) Si al 5to error se despliega un mensaje y finaliza el caso de uso.</p> <p>4.1 Si no se escoge un perfil (Ingreso).</p> <p>4.1.1 (S) Por defecto se pone un perfil básico.</p>
<p>Cuestiones:</p>

<p>Caso de Uso: Gestionar perfiles de usuarios.</p>
<p>Objetivo: Realizar el ingreso, consulta, modificación y eliminación de los perfil que le permita realizar diversas funciones del sistema.</p>
<p>Actores: Administrador(A), Sistema(S)</p>
<p>Precondiciones: Que el administrador este identificado y autenticado en el sistema. Que existan funciones creadas.</p>
<p>Postcondiciones:</p>
<p>Pasos:</p> <p>Ingreso</p> <ol style="list-style-type: none"> 1. (A) Decide ingresar un nuevo perfil al sistema. 2. (A) Ingresa la información del nuevo perfil. 3. (A) Se escoge las diferentes funciones. 4. (A) Se valida la información. 5. (A) Confirma el ingreso aceptando. 6. (S) Registra el nuevo perfil. 7. (S) Notifica el ingreso del nuevo perfil listando los perfiles <p>Consulta</p>

1. (A) Decide consultar los perfiles del sistema.
2. (S) Muestra lista o menú de opciones de búsqueda de perfiles.
3. (A) Escoge el perfil a ser consultado.
4. (S) Muestra la información del perfil.

Actualización

1. (A) Realiza pasos de Consulta.
2. (A) Se ingresa la nueva información o funciones del perfil.
3. (S) Se valida la información ingresada.
4. (S) Pide la confirmación de la actualización del perfil.
5. (A) Confirma la actualización aceptando.
6. (S) Se registra los cambios en el perfil.
7. (S) Notifica la actualización del perfil al administrador listando los perfiles.

Eliminación

1. (A) Realiza pasos de Consulta.
2. (S) Pide la confirmación de la eliminación del perfil.
3. (A) Confirma la eliminación aceptando.
4. (S) Marca al perfil como eliminado.

Extensiones:

- 2.1-3.1, 5.1, 6.1 Se cancela la acción sobre el perfil.
- 2.1.1-3.1.1, 5.1.1, 6.1.1 (S) Pide la confirmación de la cancelación.
- 2.1.2-3.1.2, 5.1.2, 6.1.2 (A) Confirma la cancelación aceptando.
- 2.1.3-3.1.3, 5.1.3, 6.1.3 (S) Finaliza el caso de uso.
- 3.1 Si no se escoge funciones para el perfil.(Ingreso)
- 3.1.1 (S) Se personaliza al perfil con funciones básicas.
- 3.1, 4.1 Los datos ingresados son incorrectos. (Ingreso y Actualización)
- 3.1.1, 4.1.1 (S) Indica el error y regresa a flujo 2.
- 3.1.2, 4.1.2 (S) Si al 5to error se despliega un mensaje y finaliza el caso de uso.

Cuestiones:

Caso de Uso: Gestionar categorías para el foro de discusión.

Objetivo: Realizar el ingreso, consulta, modificación y eliminación de las categorías para el foro de discusión.

Actores: Administrador(A), Sistema(S)
Precondiciones: Que el administrador este identificado y autenticado en el sistema.
Postcondiciones:
<p>Pasos:</p> <p>Ingreso</p> <ol style="list-style-type: none"> 1. (A) Decide ingresar una nueva categoría al foro. 2. (A) Ingresa la información de la nueva categoría. 3. (A) Se valida la información. 4. (A) Confirma el ingreso aceptando. 5. (S) Registra la nueva categoría. 6. (S) Notifica el ingreso de la nueva categoría listando las categorías. <p>Consulta</p> <ol style="list-style-type: none"> 1. (A) Decide consultar las categorías del foro. 2. (S) Muestra lista o menú de opciones de búsqueda. 3. (A) Escoge la categoría a ser consultado. 4. (S) Muestra la información de la categoría. <p>Actualización</p> <ol style="list-style-type: none"> 1. (A) Realiza pasos de Consulta. 2. (A) Se ingresa la nueva información. 3. (S) Se valida la información ingresada. 4. (A) Confirma la actualización aceptando. 5. (S) Se registra los cambios en la categoría. 6. (S) Notifica la actualización de la categoría al administrador listando las categorías. <p>Eliminación</p> <ol style="list-style-type: none"> 1. (A) Realiza pasos de Consulta. 2. (S) Pide la confirmación de la eliminación de la categoría. 3. (A) Confirma la eliminación aceptando. 4. (S) Marca la categoría como eliminada. 5. (S) Notifica la eliminación de la categoría al administrador.
<p>Extensiones:</p> <p>2.1-3.1, 5.1, 6.1 Se cancela la acción sobre las funciones de las categorías del foro de discusión.</p>

<p>2.1.1-3.1.1, 5.1.1, 6.1.1 (S) Pide la confirmación de la cancelación.</p> <p>2.1.2-3.1.2, 5.1.2, 6.1.2 (A) Confirma la cancelación aceptando.</p> <p>2.1.3-3.1.3, 5.1.3, 6.1.3 (S) Finaliza el caso de uso.</p> <p>3.1, 4.1 Los datos ingresados son incorrectos. (Ingreso y Actualización)</p> <p>3.1.1, 4.1.1 (S) Indica el error y regresa a flujo 2.</p> <p>3.1.2, 4.1.2 (S) Si al 5to error se despliega un mensaje y finaliza el caso de uso.</p>
<p>Cuestiones:</p>

<p>Caso de Uso: Gestionar funciones del sistema.</p>
<p>Objetivo: Realizar el ingreso, consulta, modificación y eliminación de las funciones que el sistema va a permitir.</p>
<p>Actores: Administrador(A), Sistema(S)</p>
<p>Precondiciones: Que el administrador este identificado y autenticado en el sistema.</p>
<p>Postcondiciones:</p>
<p>Pasos:</p> <p>Ingreso</p> <ol style="list-style-type: none"> 1. (A) Decide ingresar una nueva función del sistema. 2. (A) Ingresa la información de la nueva función. 3. (A) Se valida la información. 4. (A) Confirma el ingreso aceptando. 5. (S) Registra la nueva función. 6. (S) Notifica el ingreso de la nueva función listando las funciones. <p>Consulta</p> <ol style="list-style-type: none"> 1. (A) Decide consultar las funciones del sistema. 2. (S) Muestra lista o menú de opciones de búsqueda. 3. (A) Escoge la función a ser consultado. 4. (S) Muestra la información de la función. <p>Actualización</p> <ol style="list-style-type: none"> 1. (A) Realiza pasos de Consulta. 2. (A) Se ingresa la nueva información. 3. (S) Se valida la información ingresada.

4. (A) Confirma la actualización aceptando.
5. (S) Se registra los cambios en la función.
6. (S) Notifica la actualización de la función al administrador listando las funciones.

Eliminación

1. (A) Realiza pasos de Consulta.
2. (S) Pide la confirmación de la eliminación de la función.
3. (A) Confirma la eliminación aceptando.
4. (S) Marca la función como eliminada.
5. (S) Notifica la eliminación de la función al administrador.

Extensiones:

- 2.1-3.1, 5.1, 6.1 Se cancela la acción sobre las funciones del sistema.
- 2.1.1-3.1.1, 5.1.1, 6.1.1 (S) Pide la confirmación de la cancelación.
- 2.1.2-3.1.2, 5.1.2, 6.1.2 (A) Confirma la cancelación aceptando.
- 2.1.3-3.1.3, 5.1.3, 6.1.3 (S) Finaliza el caso de uso.
- 3.1, 4.1 Los datos ingresados son incorrectos. (Ingreso y Actualización)
- 3.1.1, 4.1.1 (S) Indica el error y regresa a flujo 2.
- 3.1.2, 4.1.2 (S) Si al 5to error se despliega un mensaje y finaliza el caso de uso.

Cuestiones:

Caso de Uso: Gestionar datos de la Organización

Objetivo: Realizar el ingreso o modificación de los datos de la organización.

Actores: Administrador(A), Sistema(S)

Precondiciones: Que el administrador este identificado y autenticado en el sistema.

Pasos:

Ingreso

1. (A) Ingresa los datos de la empresa.
2. (A) Se valida la información de la empresa.
3. (A) Confirma la actualización aceptando.
4. (S) Registra la información de la organización.
5. (S) Notifica la actualización de la empresa.

Actualización

1. (A) Se ingresa nueva información para la actualización de los datos de la organización.
2. (S) Se valida la información.
3. (S) Pide la confirmación de la actualización de la empresa.
4. (A) Confirma la actualización aceptando.
5. (S) Registra la información de la organización.
6. (S) Notifica la actualización de la empresa.

Extensiones:

1.1-4.1 Se cancela la acción a ser efectuada sobre los datos de la empresa.

1.1.1-4.1.1 (S) Pide la confirmación de la cancelación.

1.1.2-4.1.2 (A) Confirma la cancelación aceptando.

1.1.3-4.1.3 (S) Finaliza el caso de uso.

2.1 Los datos ingresados son incorrectos.

2.1.1 (S) Indica el error y regresa a flujo 1.

2.1.2 (S) Si al 5to error se despliega un mensaje y finaliza el caso de uso.

Cuestiones:

Caso de Uso: Gestionar tareas del usuario.

Objetivo: Realizar el ingreso, consulta, actualización y eliminación de tareas del usuario.

Actores: Usuario(U), Sistema(S)

Precondiciones: Que el usuario este identificado y autenticado en el sistema. Que al menos exista un usuario en la base de datos. Que exista por lo menos una tarea. (En el caso de Consultar, Actualización y Eliminación)

Pasos:

Ingreso

1. (U) Quiere crear una nueva tarea.
2. (U) Ingresa los datos de la tarea (Nombre de la tarea, Descripción, Hora, Fecha de ingreso, fecha de inicio, fecha de finalización, porcentaje de avance, estado actual).
3. (S) Valida los datos de la tarea.

4. (U) Confirma el ingreso aceptando.
5. (S) Registra la tarea y se asigna una identificación.

Consultar

1. (U) Decide consultar las tareas en el sistema.
2. (S) Muestra lista o menú de opciones de búsqueda de las tareas.
3. (U) Escoge la tarea a ser consultada.
4. (S) Muestra la información de la tarea.

Actualizar

1. (U) Realiza pasos de Consulta.
2. (U) Edita o ingresa información de la tarea.
3. (S) Valida los datos.
4. (U) Confirma la actualización aceptando.
5. (S) Registra los cambios en la información de la tarea.
6. (S) Notifica la actualización de la tarea al usuario.

Eliminar

1. (U) Realiza pasos de Consulta.
2. (U) Escoge la opción de Eliminación.
3. (U) Confirma la eliminación aceptando.
4. (S) Marca a la tarea como finalizada.
5. (S) Notifica la eliminación de la tarea al usuario.

Extensiones:

- 2.1-4.1, 5.1 Se cancela la acción sobre la gestión de tareas.
 - 2.1.1-4.1.1, 5.1.1 (S) Pide la confirmación de la cancelación.
 - 2.1.2-4.1.2, 5.1.2 (U) Confirma la cancelación aceptando.
 - 2.1.3-4.1.3, 5.1.3 (S) Finaliza el caso de uso.
- 3.1 Los datos ingresados son incorrectos. (Ingreso y Actualización)
 - 3.1.1 (S) Indica el error y regresa a flujo 2.
 - 3.1.2 (S) Si al 5to error se despliega un mensaje y finaliza el caso de uso.

Cuestiones:

Caso de Uso: Enviar Mensajes Instantáneos en tiempo real.
Objetivo: Enviar Mensajes en tiempo real a usuarios que estén utilizando el sistema.
Actores: Usuario(U1,U2), Sistema(S)
Precondiciones: Que el usuario este identificado y autenticado en el sistema. Que al menos exista dos usuarios en la base de datos.
<p>Pasos:</p> <ol style="list-style-type: none"> 1. (U1) Decide entablar una sesión de mensajes instantáneos en tiempo real con otro usuario (U2). 2. (S) Muestra una lista de los usuarios en línea con el sistema 3. (U1) Escoge al usuario con el cual quiere establecer una sesión de mensajes en tiempo real. 4. (S) Activa la pantalla para el envío de mensajes. 5. (U1) Se procede a escribir el mensaje. 6. (U1) Se procede a enviar el mensaje. 7. (S) Transmite el mensaje a otro usuario (U2). 8. (U2) Acepta el comenzar una sala de Chat con el usuario(U1) 9. (U2) Responde al mensaje.
<p>Extensiones:</p> <ol style="list-style-type: none"> 2.1-9.1 Se cancela la invitación a empezar mensajería instantánea. <ol style="list-style-type: none"> 2.1.1-9.1.1 (S) Pide la confirmación de la cancelación. 2.1.2-9.1.2 (U) Confirma la cancelación aceptando. 2.1.3-9.1.3 (S) Finaliza el caso de uso. 3.1 Si se escoge a un usuario equivocado. <ol style="list-style-type: none"> 3.1.1 (U) El usuario puede confirmar el cierre de la sesión para conversar por mensajes instantáneos. 3.1.2 (S) Procede a cancelar el inicio de la sala de Chat entre las dos personas. 3.1.3 (S) Regresa a 2. 6.1 Si el mensaje no puede ser entregado a su destinatario. <ol style="list-style-type: none"> 6.1.1 (S) El mensaje se encola hasta su posterior reenvió. 6.1.2 (S) Si el mensaje no se puede enviar se notifica a su remitente.
Cuestiones:

Caso de Uso: Categorizar a Usuarios en grupos
Objetivo: Crear una categoría que este conformada por uno o varios usuarios.
Actores: Usuario(U), Sistema(S)
Precondiciones: Que el usuario este identificado y autenticado en el sistema. Que al menos exista un usuario en la base de datos.
<p>Pasos:</p> <p>Ingreso</p> <ol style="list-style-type: none"> 1. (U) Decide ingresar un nuevo grupo al sistema. 2. (U) Ingresa la información del nuevo grupo. 3. (U) Realiza pasos de consulta en caso de uso "Gestión de Usuarios". 4. (U) Escoge el usuario o usuarios. 5. (U) Se valida la información. 6. (U) Confirma el ingreso aceptando. 7. (S) Registra el nuevo grupo. 8. (S) Notifica el ingreso del nuevo grupo al usuario. <p>Consulta</p> <ol style="list-style-type: none"> 1. (U) Decide consultar un grupo registrado en el sistema. 2. (S) Muestra lista o la opción de búsqueda de grupos(s). 3. (U) Escoge el grupo a ser consultado. 4. (S) Se presenta la información del grupo. <p>Actualización</p> <ol style="list-style-type: none"> 1. (U) Decide modificar un grupo del sistema. 2. (U) Realiza los pasos de consulta. 3. (U) Edita la información y los usuarios pertenecientes al grupo. 4. (U) Se valida la información. 5. (U) Confirma la actualización aceptando. 6. (S) Registra la actualización del grupo. <p>Eliminación</p> <ol style="list-style-type: none"> 1. (U) Decide eliminar un grupo del sistema. 2. (U) Realiza los pasos de consulta. 3. (U) Selecciona la eliminación del grupo. 4. (U) Confirma la eliminación aceptando. 5. (S) Registra la eliminación del grupo.

Extensiones:

- 2.1-5.1, 6.1, 7.1 Se cancela la acción sobre la gestión de un grupo.
 - 2.1.1-5.1.1, 6.1.1, 7.1.1 (S) Pide la confirmación de la cancelación.
 - 2.1.2-5.1.2, 6.1.2, 7.1.2 (U) Confirma la cancelación aceptando.
 - 2.1.3-5.1.3, 6.1.3, 7.1.3 (S) Finaliza el caso de uso.
- 4.1, 5.1 Los datos ingresados son incorrectos.
 - 4.1.1, 5.1.1 (S) Indica el error y regresa a flujo 2.
 - 4.1.2, 5.1.2 (S) Si al 5to error se despliega un mensaje y finaliza el caso de uso.

Cuestiones:

Caso de Uso: Ingresar a Pizarra virtual.

Objetivo: Permitir la utilización de la pizarra virtual con otros usuarios o grupos.

Actores: Usuario(U1), Usuarios(Ux), Sistema(S)

Precondiciones: Que el usuario este identificado y autenticado en el sistema. Que al menos exista un usuario en la base de datos.

Pasos:

1. (U1) Decide entablar una sesión de pizarra virtual en tiempo real con otro(s) usuario(s) (Ui).
2. (S) Muestra una lista de los usuarios en línea con el sistema.
3. (U1) Escoge al usuario o grupo con el cual quiere establecer una sesión de mensajes en tiempo real.
4. (S) Activa la pantalla de pizarra virtual.
5. (U1) Se procede a escribir sobre la pizarra.
6. (S) Refleja lo que el usuario esta escribiendo a las demás pizarras.
7. (Ux) Acepta el comenzar pizarra virtual con el usuario(Ux)

Extensiones:

- 2.1-7.1 Se cancela la invitación a empezar pizarra virtual.
 - 2.1.1-7.1.1 (S) Pide la confirmación de la cancelación.
 - 2.1.2-7.1.2 (U) Confirma la cancelación aceptando.
 - 2.1.3-7.1.3 (S) Finaliza el caso de uso.
- 3.1 Si se escoge a un usuario equivocado.
 - 3.1.1 (U) El usuario puede confirmar el cierre de la sesión para el uso

<p>de la pizarra virtual.</p> <p>3.1.2 (S) Procede a cancelar el inicio de la pizarra virtual.</p> <p>3.1.3 (S) Regresa a 2.</p> <p>6.1 Si no se puede reflejar la información al destinatario.</p> <p>6.1.1 (S) Se envía un mensaje a su remitente indicando a las personas que no les llegó la información en pizarra.</p>
Cuestiones:

Caso de Uso: Gestionar foros de discusión.
Objetivo: Que un usuario ingrese, consulte, actualice, responda y elimine un tema del foro de discusión.
Actores: Usuario(U), Sistema(S)
<p>Precondiciones: Que el usuario este identificado y autenticado en el sistema.</p> <p>Que al menos exista un usuario en la base de datos.</p> <p>Para la actualización o eliminación deberá ser el propietario del tema.</p> <p>Que al menos exista un tema registrado en la base de datos.(Consulta, Actualización y Eliminación)</p>
<p>Pasos:</p> <p>Ingreso</p> <ol style="list-style-type: none"> 1. (U) Quiere crear un nuevo tema en el foro de discusión. 2. (U) Ingresa los datos del tema (Titulo, Descripción, Tema, fecha de inicio del tema, fecha de caducidad, Censura del tema). 3. (U) Escoge la categoría del tema. 4. (S) Valida los datos. 5. (U) Confirma el ingreso aceptando. 6. (S) Registra tema y se asigna una identificación. <p>Consulta</p> <ol style="list-style-type: none"> 1. (U) Decide consultar los temas registrados en el sistema. 2. (S) Muestra lista o menú de opciones de búsqueda de temas. 3. (U) Escoge el tema a ser consultado. 4. (S) Muestra la información del tema. <p>Actualización</p> <ol style="list-style-type: none"> 1. (U) Decide modificar o actualizar a un tema registrado en el sistema. 2. (S) Realiza los pasos de consulta.

3. (U) Edita o ingresa información en el tema.
4. (S) Valida los datos.
5. (U) Confirma la actualización aceptando.
6. (S) Registra los cambios en la información del tema.
7. (S) Notifica la actualización del usuario.

Eliminación

1. (U) Decide eliminar un tema registrado en el sistema.
2. (U) Realiza los pasos de consulta.
3. (U) Confirma la eliminación aceptando.
4. (S) Notifica la eliminación al usuario.

Responder a tema

1. (U) Decide consultar los temas registrados en el sistema.
2. (U) Realiza los pasos de consulta.
3. (U) Se ingresa la respuesta al tema.
4. (S) Valida los datos.
5. (U) Confirma la actualización aceptando.
6. (S) Registra los cambios en las respuestas del tema.

Extensiones:

- 2.1-5.1 Se cancela cualquier acción sobre la gestión del foro de discusión.
 - 2.1.1-5.1.1 (S) Pide la confirmación de la cancelación.
 - 2.1.2-5.1.2 (U) Confirma la cancelación aceptando.
 - 2.1.3-5.1.3 (S) Finaliza el caso de uso.
- 4.1 Los datos ingresados son incorrectos. (Ingreso, Actualización y Respuesta a un tema)
 - 4.1.1 (S) Indica el error y regresa a flujo 2.
 - 4.1.2 (S) Si al 5to error se despliega un mensaje y finaliza el caso de uso.

Cuestiones:

Caso de Uso: Realizar ingreso al sistema.

Objetivo: Que el usuario pueda ser validado y autenticado para hacer uso del sistema.

Actores: Sistema(S) Usuario(U)

Precondiciones: Que al menos exista un usuario en la base de datos que no sea el administrador.

Pasos:

1. (U) Decide ingresar al sistema.
2. (S) Presenta dispositivo visual de acceso al sistema.
3. (U) Ingresa su clave de acceso y contraseña.
4. (S) Verifica los datos.
5. (S) Presenta el menú y asigna una sesión al usuario.

Extensiones:

- 4.1 Que no se encuentre el usuario.
 - 4.1.1 (S) Genera un error y lo registra.
 - 4.1.2 (S) Envía un mensaje a pantalla informando que fue fallido la verificación.
 - 4.1.3 (S) Finaliza el caso de uso.

Cuestiones:

2.2.- ANÁLISIS

2.2.1.- MODELO DE ANÁLISIS

En el modelo de análisis para el Sistema de Colaboración se va a estructurar los requisitos funcionales del sistema para su mejor comprensión a través de clases de análisis.

2.2.1.2.- Diagramas de Clase

Los diagramas de clase se representaran en conjunto con los diagramas de colaboración para cada caso de uso.

2.2.1.3.- Diagramas de Colaboración

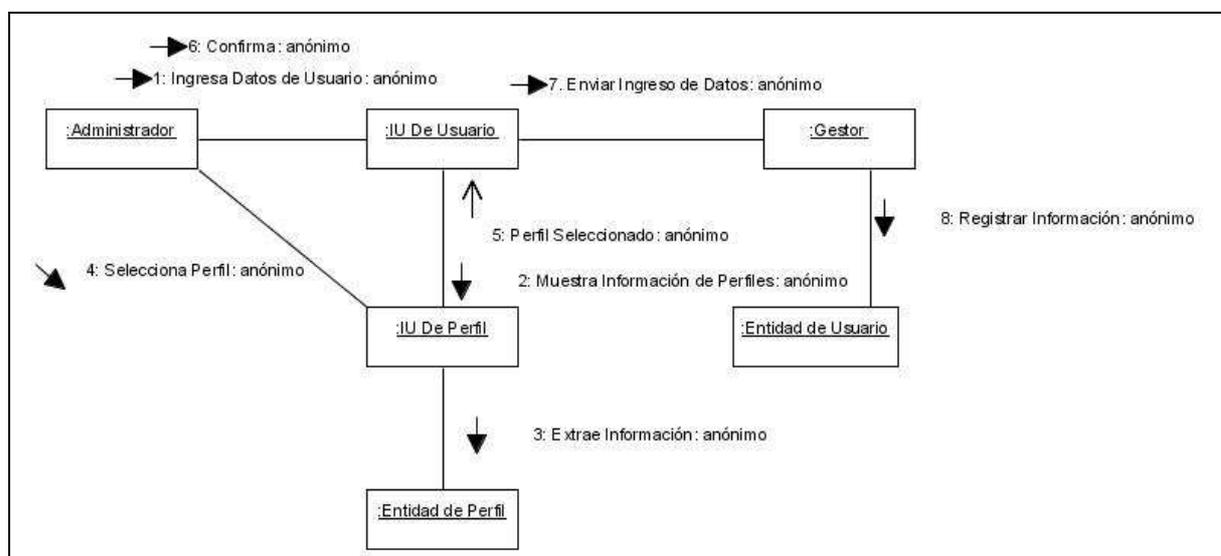
Son diagramas que representan las interacciones que tienen los diferentes objetos que se lo representan a través de clases.

Cada caso de uso deberá constar de un modelo de colaboración que represente las diferentes clases que participen en la funcionalidad del caso de uso y que sea identificadas en el mismo.

Para realizar los diagramas de colaboración que intervienen en el presente proyecto se realiza primero una descripción en la que consta el modelo al que corresponde, el diagrama a cual hace referencia, el caso de uso al que hace referencia, las diferentes clases y sus respectivos estereotipos, la figura a la que representa y por ultimo el flujo de sucesos al cual hace referencia la descripción.

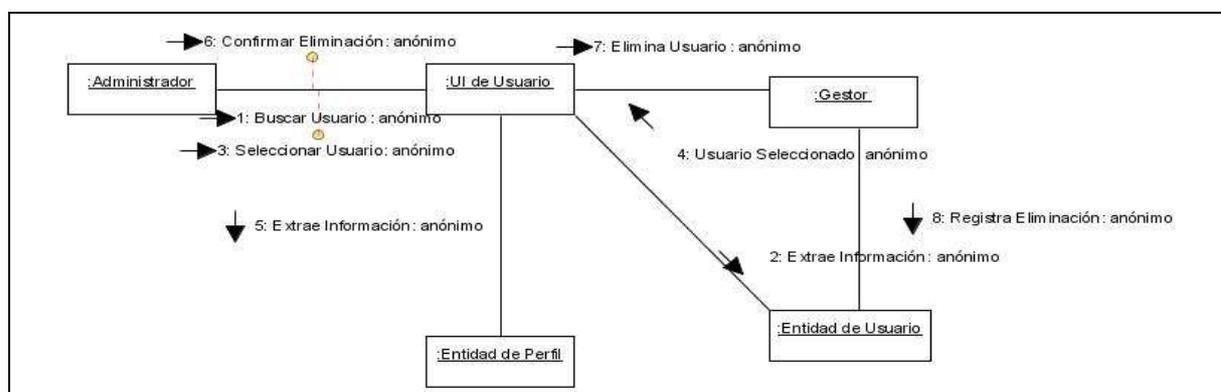
Todos los diagramas de colaboración figuras 2.4 a la 2.37 que siguen a continuación fueron realizadas con la herramienta Poseidón para UML y con referencia al autor de este proyecto.

MODELO:	Modelo de Análisis
DIAGRAMA:	Diagrama de Colaboración
CASO DE USO:	Gestión de Usuarios – Ingreso de Usuario
CLASES DE INTERFAZ:	Interfaz de Perfil
	Interfaz de Usuario
CLASE DE CONTROL:	Gestor
CLASES DE ENTIDAD:	Entidad de Usuario
	Entidad de Perfil
FIGURA:	Figura 2.4
FLUJO DE SUCECOS	
<ol style="list-style-type: none"> 1.- El administrador ingresa los datos. 2.- Se procede a mostrar los perfiles. 3.- Extrae la información de perfiles. 4.- Selecciona un perfil. 5.- Se procede a registrar el perfil con los datos. 6.- Confirma el envío de los datos. 7.- Se envía el ingreso de los datos hacia el gestor. 8.- Se registra los datos. 	



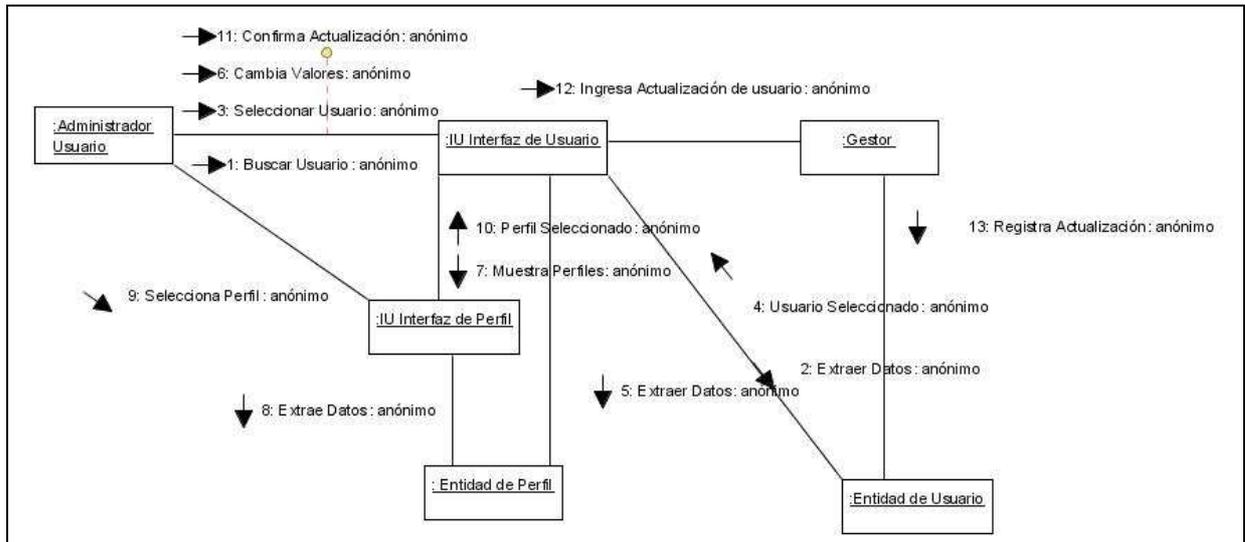
[Figura 2.4]

MODELO:	Modelo de Análisis
DIAGRAMA:	Diagrama de Colaboración
CASO DE USO:	Gestión de Usuarios – Eliminación de Usuario
CLASES DE INTERFAZ:	Interfaz de Administrador
CLASES DE CONTROL:	Gestor
CLASES DE ENTIDAD:	Entidad de Usuario
	Entidad de Perfil
FIGURA:	Figura 2.5
FLUJO DE SUCESOS	
<ol style="list-style-type: none"> 1.- El administrador busca un usuario. 2.- Extrae la información de usuarios. 3.- Selecciona un usuario. 4.- Se extrae la información del usuario. 5.- Se extrae la información del perfil del usuario. 6.- Se confirma la eliminación. 7.- Se envía la orden de eliminación del usuario hacia el gestor. 8.- Se registra el usuario como eliminado. 	



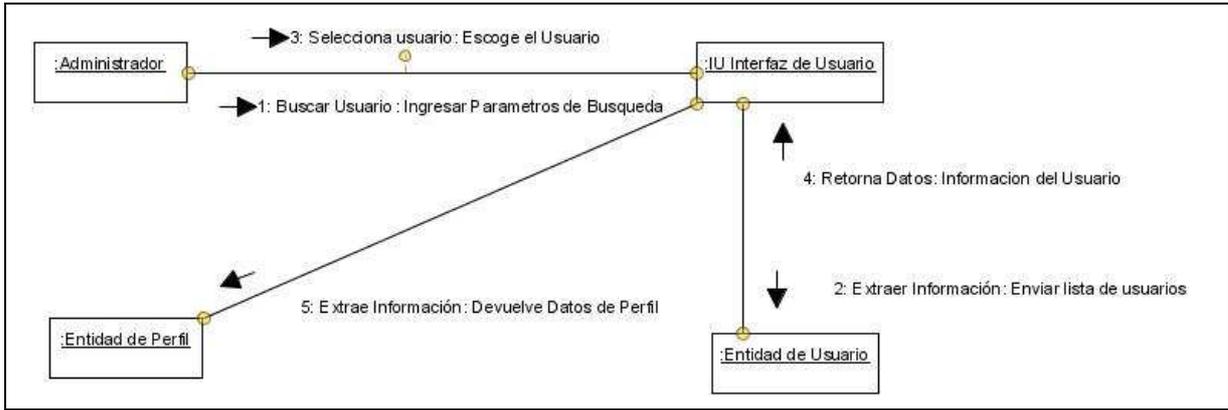
[Figura 2.5]

MODELO:	Modelo de Análisis
DIAGRAMA:	Diagrama de Colaboración
CASO DE USO:	Gestión de Usuarios - Actualización
CLASES DE INTERFAZ:	Interfaz de Perfil
	Interfaz de Administrador
CLASES DE CONTROL:	Gestor
CLASES DE ENTIDAD:	Entidad de Usuario
	Entidad de Perfil
FIGURA:	Figura 2.6
FLUJO DE SUCESOS	
<ol style="list-style-type: none"> 1.- El administrador busca un usuario. 2.- Extrae la información de usuarios. 3.- Selecciona un usuario. 4.- Se extrae la información del usuario. 5.- Se extrae la información del perfil del usuario. 6.- Cambia Valores. 7.- Muestra perfiles. 8.- Extrae los datos de perfiles. 9.- Selecciona un perfil. 10.- Perfil Seleccionado 11.- Confirma Actualización 12.- Ingresa Actualización a Gestor 13.- Registra Actualización 	



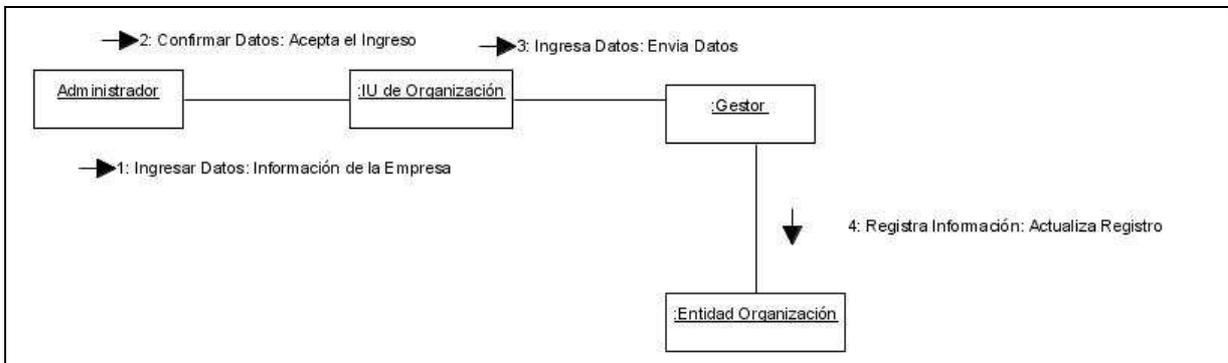
[Figura 2.6]

MODELO:	Modelo de Análisis
DIAGRAMA:	Diagrama de Colaboración
CASO DE USO:	Gestión de Usuarios - Consulta de Usuario
CLASES DE INTERFAZ:	Interfaz de Administrador y Usuario
CLASES DE CONTROL:	
CLASES DE ENTIDAD:	Entidad de Usuario
	Entidad de Perfil
FIGURA:	Figura 2.7
FLUJO DE SUCECOS	
<ol style="list-style-type: none"> 1.- El administrador busca un usuario. 2.- Extrae la información de usuarios. 3.- Selecciona un usuario. 4.- Se extrae la información del usuario. 5.- Se extrae la información del perfil del usuario. 	



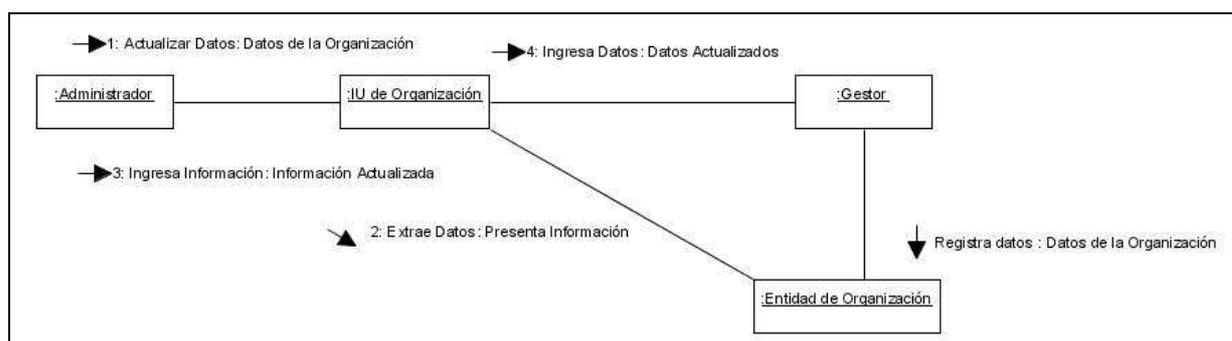
[Figura 2.7]

MODELO:	Modelo de Análisis
DIAGRAMA:	Diagrama de Colaboración
CASO DE USO:	Gestionar organización - Ingresar datos Organización
CLASES DE INTERFAZ:	Interfaz de Organización
CLASES DE CONTROL:	Gestor
CLASES DE ENTIDAD:	Entidad de Organización
FIGURA:	Figura 2.8
FLUJO DE SUCESOS	
<ol style="list-style-type: none"> 1.- El administrador ingresa datos de la organización. 2.- Confirma el Ingreso de los datos. 3.- Ingresar datos. 4.- Se registra los datos. 	



[Figura 2.8]

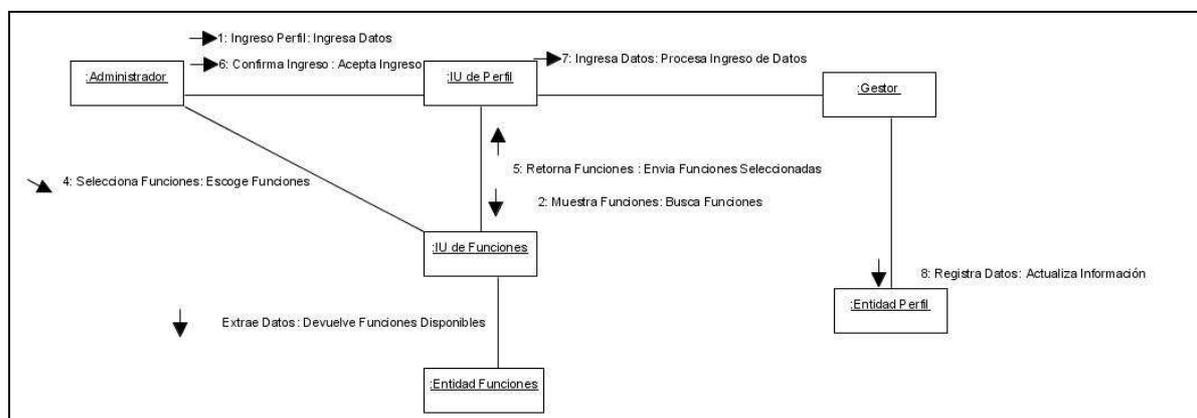
MODELO:	Modelo de Análisis
DIAGRAMA:	Diagrama de Colaboración
CASO DE USO:	Gestionar organización - Actualizar datos Organización
CLASES DE INTERFAZ:	Interfaz de Organización
CLASES DE CONTROL:	Gestor
CLASES DE ENTIDAD:	Entidad de Organización
FIGURA:	Figura 2.9
FLUJO DE SUCESOS	
<ol style="list-style-type: none"> 1.- El administrador datos de la organización. 2.- Confirma el Ingreso de los datos. 3.- Ingresa datos. 4.- Se registra los datos. 	



[Figura 2.9]

MODELO:	Modelo de Análisis
DIAGRAMA:	Diagrama de Colaboración
CASO DE USO:	Gestión de perfiles -Ingreso de Perfil
CLASES DE INTERFAZ:	Interfaz de Perfil
	Interfaz de Funciones
CLASES DE CONTROL:	Gestor
CLASES DE ENTIDAD:	Entidad de Perfil
	Entidad de Funciones

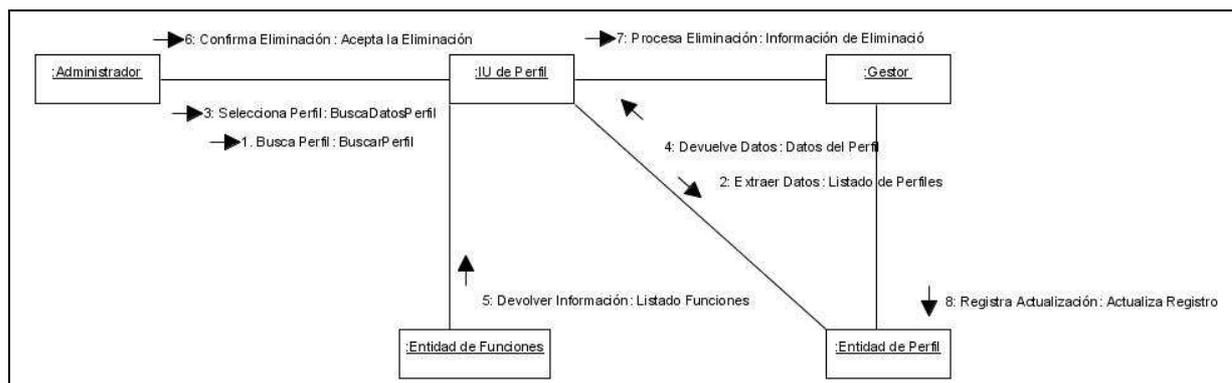
FIGURA:	Figura 2.10
FLUJO DE SUCESOS	
<ol style="list-style-type: none"> 1.- El administrador ingresa los datos. 2.- Se procede a mostrar las funciones. 3.- Extrae la información de funciones. 4.- Selecciona una o varias funciones. 5.- Retorna funciones escogidas. 6.- Confirma el envío de los datos. 7.- Se envía el ingreso de los datos hacia el gestor. 8.- Se registra los datos. 	



[Figura 2.10]

MODELO:	Modelo de Análisis
DIAGRAMA:	Diagrama de Colaboración
CASO DE USO:	Gestión de perfiles - Eliminación de Perfil
CLASES DE INTERFAZ:	Interfaz de Perfil
CLASES DE CONTROL:	Gestor
CLASES DE ENTIDAD:	Entidad de Funciones
	Entidad de Perfil
FIGURA:	Figura 2.11
FLUJO DE SUCESOS	
1.- El administrador busca un perfil.	

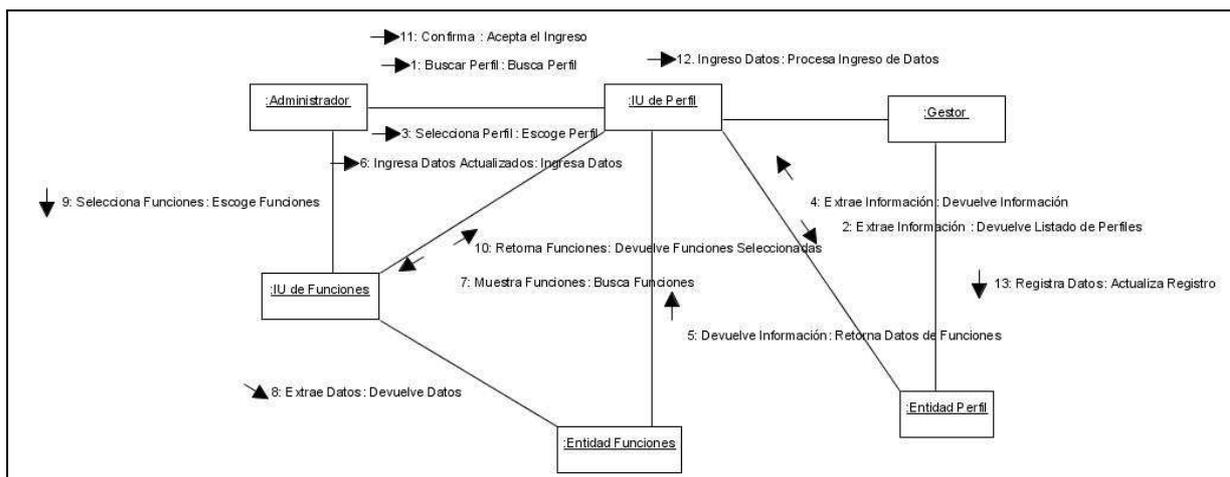
- 2.- Extrae la información como resultado de la búsqueda.
- 3.- Selecciona un perfil.
- 4.- Se extrae la información del perfil seleccionado.
- 5.- Se confirma la eliminación.
- 6.- Se envía la orden de eliminación del usuario hacia el gestor.
- 7.- Se registra el perfil como eliminado.



[Figura 2.11]

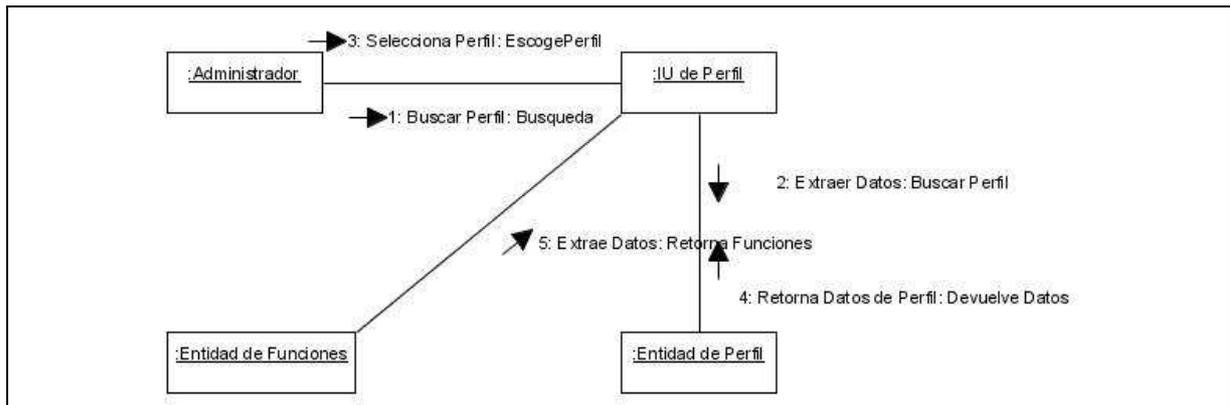
MODELO:	Modelo de Análisis
DIAGRAMA:	Diagrama de Colaboración
CASO DE USO:	Gestión de perfiles - Actualización de Perfil
CLASES DE INTERFAZ:	Interfaz de Perfil Interfaz de Funciones
CLASES DE CONTROL:	Gestor
CLASES DE ENTIDAD:	Entidad de Usuario Entidad de Funciones
FIGURA:	Figura 2.12
FLUJO DE SUCESOS	
<ol style="list-style-type: none"> 1.- El administrador busca un perfil. 2.- Extrae la información como resultado de la búsqueda. 3.- Selecciona un perfil. 4.- Se extrae la información del perfil seleccionado. 5.- Se extrae la información de las funciones. seleccionadas. 6.- Ingresa Actualización de los datos. 7.- Muestra Lista de Funciones. 	

- 8.- Extrae la información de las funciones
- 9.- Selecciona una o varias funciones.
- 10.- Retorna funciones escogidas.
- 11.- Confirma el envío de los datos.
- 12.- Se envía el ingreso de los datos hacia el gestor.
- 13.- Se registra los datos.



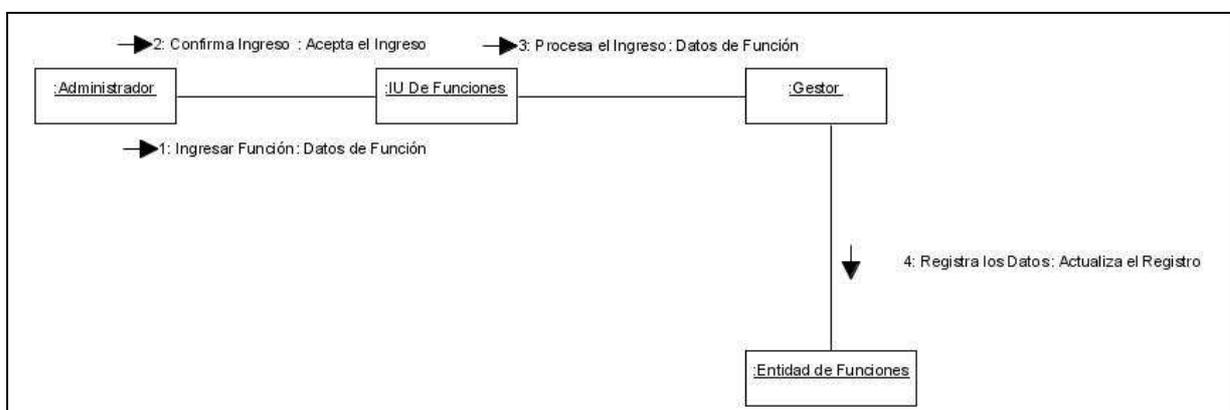
[Figura 2.12]

MODELO:	Modelo de Análisis
DIAGRAMA:	Diagrama de Colaboración
CASO DE USO:	Gestión de perfiles - Búsqueda de Perfil
CLASES DE INTERFAZ:	Interfaz de Perfil
CLASES DE CONTROL:	
CLASES DE ENTIDAD:	Entidad de Funciones
	Entidad de Perfil
FIGURA:	Figura 2.13
FLUJO DE SUCESOS	
<ol style="list-style-type: none"> 1.- El administrador busca un perfil. 2.- Extrae la información como resultado de la búsqueda. 3.- Selecciona un perfil. 4.- Se extrae la información del perfil seleccionado. 5.- Se extrae la información de las funciones del perfil. 	



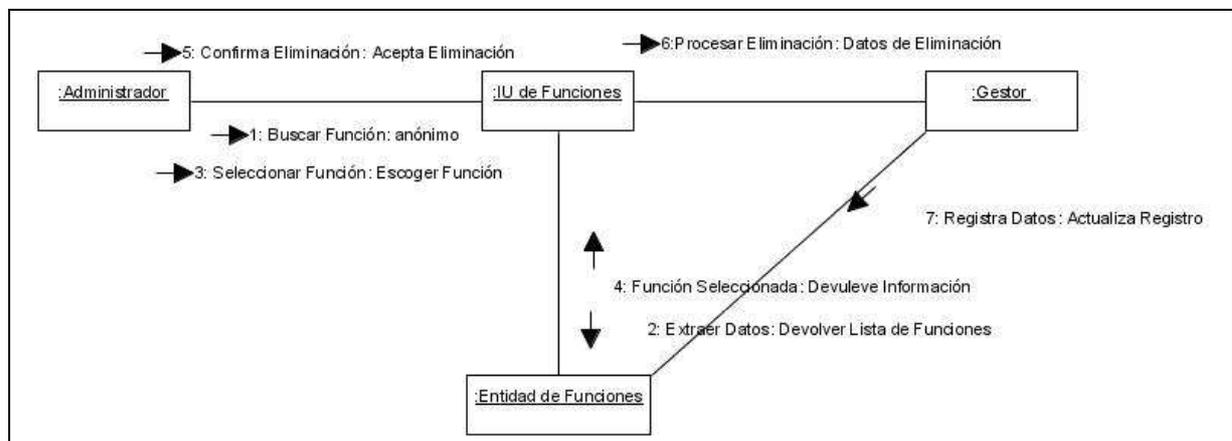
[Figura 2.13]

MODELO:	Modelo de Análisis
DIAGRAMA:	Diagrama de Colaboración
CASO DE USO:	Gestión de Funciones - Ingreso de Funciones
CLASES DE INTERFAZ:	Interfaz de Funciones
CLASES DE CONTROL:	Gestor
CLASES DE ENTIDAD:	Entidad de Funciones
FIGURA:	Figura 2.14
FLUJO DE SUCEOS	
1.- El administrador ingresa los datos.	
2.- Confirma el envío de los datos.	
3.- Se envía el ingreso de los datos hacia el gestor.	
4.- Se registra los datos.	



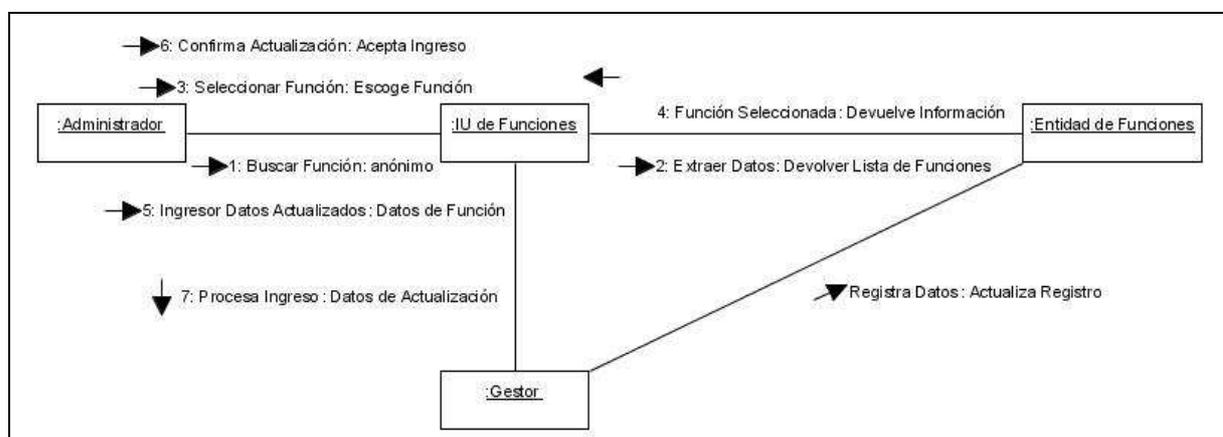
[Figura 2.14]

MODELO:	Modelo de Análisis
DIAGRAMA:	Diagrama de Colaboración
CASO DE USO:	Gestión de Funciones - Eliminar Función
CLASES DE INTERFAZ:	Interfaz de Funciones
CLASES DE CONTROL:	Gestor
CLASES DE ENTIDAD:	Entidad de Funciones
FIGURA:	Figura 2.15
FLUJO DE SUCEOS	
<ol style="list-style-type: none"> 1.- El administrador busca una función. 2.- Extrae la información de la búsqueda de funciones. 3.- Selecciona una función. 4.- Se extrae la información de la función. 5.- Se confirma la eliminación. 6.- Se envía la orden de eliminación de la función hacia el gestor. 7.- Se registra la función como eliminada. 	



[Figura 2.15]

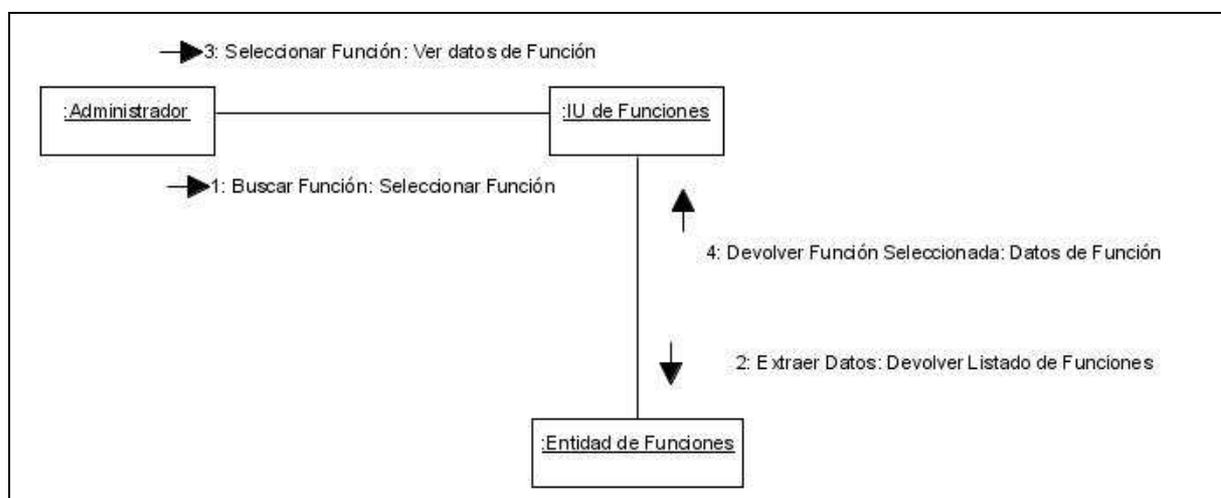
MODELO:	Modelo de Análisis
DIAGRAMA:	Diagrama de Colaboración
CASO DE USO:	Gestión de Funciones - Actualizar Función
CLASES DE INTERFAZ:	Interfaz de Funciones
CLASES DE CONTROL:	Gestor
CLASES DE ENTIDAD:	Entidad de Funciones
FIGURA:	Figura 2.16
FLUJO DE SUCEOS	
<ol style="list-style-type: none"> 1.- El administrador busca una función. 2.- Extrae la información de la búsqueda de funciones. 3.- Selecciona una función. 4.- Se extrae la información de la función. 5.- Se ingresa actualización de datos. 6.- Se confirma la actualización. 7.- Se envía la orden de actualización de la función hacia el gestor. 8.- Se registra la actualización de los datos. 	



[Figura 2.16]

MODELO:	Modelo de Análisis
DIAGRAMA:	Diagrama de Colaboración
CASO DE USO:	Gestión de Funciones - Búsqueda de

	Función
CLASES DE INTERFAZ:	Interfaz de Funciones
CLASES DE CONTROL:	Gestor
CLASES DE ENTIDAD:	Entidad de Funciones
FIGURA:	Figura 2.17
FLUJO DE SUCECOS	
<p>1.- El administrador busca una función.</p> <p>2.- Extrae la información de la búsqueda de funciones.</p> <p>3.- Selecciona una función.</p> <p>4.- Se extrae la información de la función.</p>	

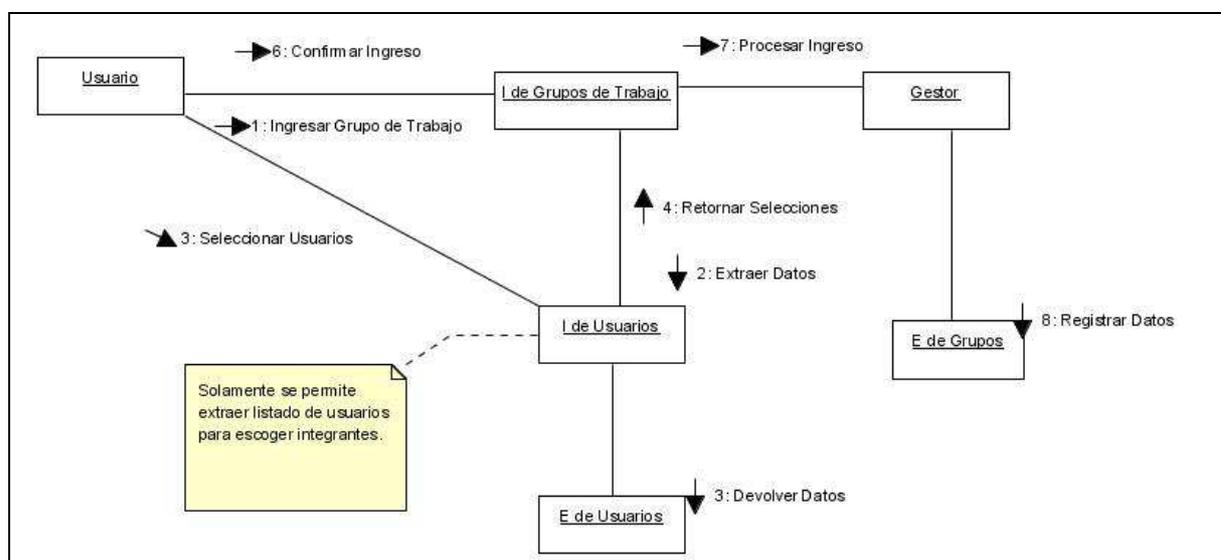


[Figura 2.17]

MODELO:	Modelo de Análisis
DIAGRAMA:	Diagrama de Colaboración
CASO DE USO:	Gestión de Grupos de Trabajo - Crear Grupo de Usuarios
CLASES DE INTERFAZ:	Interfaz de Grupos de Trabajo
	Interfaz de usuarios
CLASES DE CONTROL:	Gestor
CLASES DE ENTIDAD:	Entidad de Grupos
	Entidad de Usuarios
FIGURA:	Figura 2.18

FLUJO DE SUCESOS

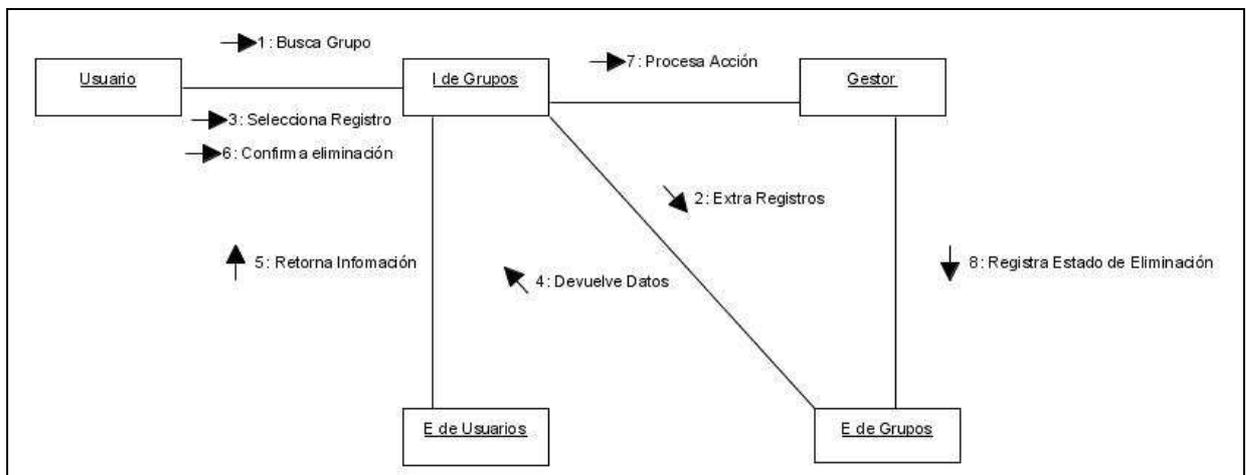
- 1.- El usuario ingresa un nuevo grupo.
- 2.- Extrae la información de la búsqueda de usuarios.
- 3.- Se devuelve los datos a la interfaz.
- 4.- Se selecciona a los usuarios.
- 5.- Se devuelve los usuarios seleccionados.
- 6.- Se confirma el ingreso.
- 7.- Se envía la orden de ingreso del grupo hacia el gestor.
- 8.- Se registra los datos.



[Figura 2.18]

MODELO:	Modelo de Análisis
DIAGRAMA:	Diagrama de Colaboración
CASO DE USO:	Gestión de Grupos de Trabajo - Eliminar Grupo de Usuarios
CLASES DE INTERFAZ:	Interfaz de Grupos de Trabajo
CLASES DE CONTROL:	Gestor
CLASES DE ENTIDAD:	Entidad de Grupos
	Entidad de Usuarios
FIGURA:	Figura 2.19
FLUJO DE SUCESOS	

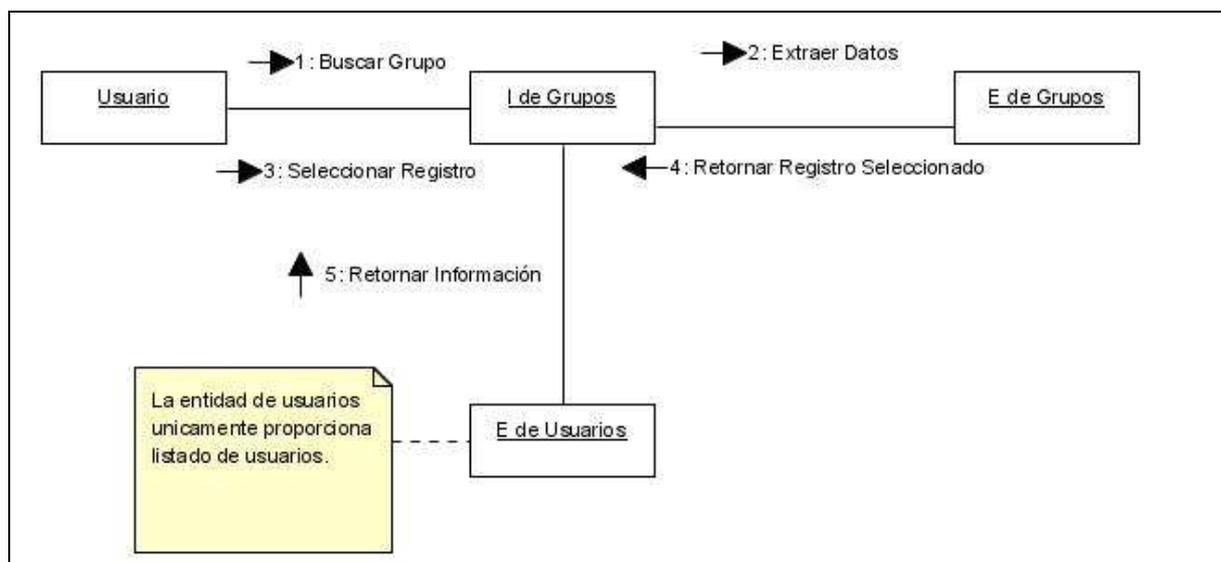
- 1.- El usuario busca un grupo.
- 2.- Extrae la información de la búsqueda de grupos.
- 3.- Se selecciona un grupo.
- 4.- Se devuelve datos del grupo.
- 5.- Se devuelve los usuarios del grupo.
- 6.- Se confirma la eliminación.
- 7.- Se envía la orden de eliminación del grupo hacia el gestor.
- 8.- Se registra la eliminación.



[Figura 2.19]

MODELO:	Modelo de Análisis
DIAGRAMA:	Diagrama de Colaboración
CASO DE USO:	Gestión de Grupos de Trabajo - Buscar Grupo de Trabajo.
CLASES DE INTERFAZ:	Interfaz de Grupos de Trabajo
CLASES DE CONTROL:	
CLASES DE ENTIDAD:	Entidad de Grupos
	Entidad de Usuarios
FIGURA:	Figura 2.20
FLUJO DE SUCESOS	

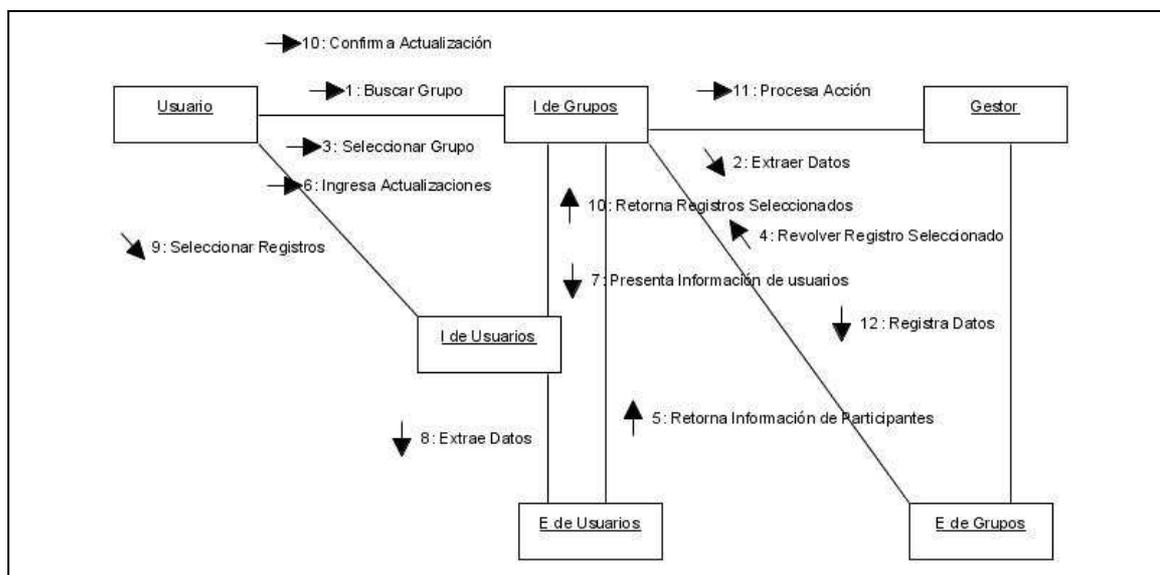
- 1.- El usuario busca un grupo.
- 2.- Extrae la información de la búsqueda de grupos.
- 3.- Se selecciona un grupo.
- 4.- Se devuelve datos del grupo.
- 5.- Se devuelve los usuarios del grupo.



[Figura 2.20]

MODELO:	Modelo de Análisis
DIAGRAMA:	Diagrama de Colaboración
CASO DE USO:	Gestión de Grupos de Trabajo - Actualizar Grupo de Trabajo.
CLASES DE INTERFAZ:	Interfaz de Grupos de Trabajo
	Interfaz de Usuarios
CLASES DE CONTROL:	Gestor
CLASES DE ENTIDAD:	Entidad de Grupos
	Entidad de Usuarios
FIGURA:	Figura 2.21
FLUJO DE SUCESOS	
<ol style="list-style-type: none"> 1.- El usuario busca un grupo. 2.- Extrae la información de la búsqueda de grupos. 3.- Se selecciona un grupo. 	

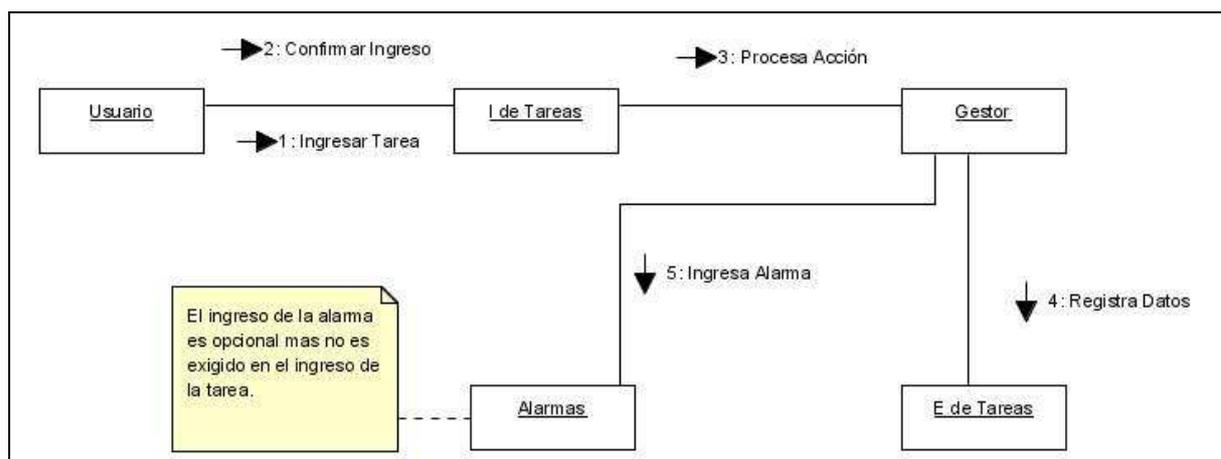
- 4.- Se devuelve datos del grupo.
- 5.- Se devuelve los usuarios del grupo.
- 6.- Se ingresa las respectivas actualizaciones.
- 7.- Se presenta los usuarios.
- 8.- Se extrae los datos para la interfaz.
- 9.- El usuario selecciona los usuarios nuevos o los quita.
- 10.- El usuario confirma la actualización.
- 11.- Se envía la orden de actualización del grupo hacia el gestor.
- 12.- Se registra los datos.



[Figura 2.21]

MODELO:	Modelo de Análisis
DIAGRAMA:	Diagrama de Colaboración
CASO DE USO:	Gestión de Tareas, Eventos o Citas - Crear nueva tarea, evento o cita.
CLASES DE INTERFAZ:	Interfaz de Tareas, Eventos o Citas.
CLASES DE CONTROL:	Gestor
CLASES DE ENTIDAD:	Entidad de Tareas
	Entidad de Alarmas
FIGURA:	Figura 2.22

FLUJO DE SUCESOS
1.- El usuario ingresa una nueva tarea, evento o cita.
2.- Se confirma el ingreso.
3.- Se envía la orden de ingreso de la nueva tarea hacia el gestor.
4.- Se registra los datos.
5.- Se envía a registrar alarma. (Opcional)



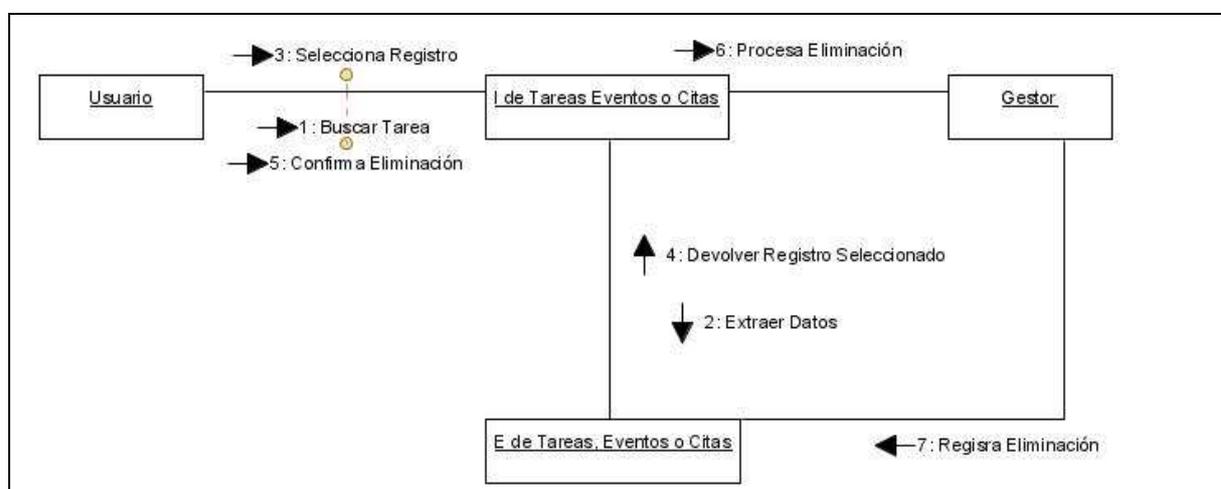
[Figura 2.22]

MODELO:	Modelo de Análisis
DIAGRAMA:	Diagrama de Colaboración
CASO DE USO:	Gestión de Tareas, Eventos o Citas – Consultar tarea, evento o cita.
CLASES DE INTERFAZ:	Interfaz de Tareas, Eventos o Citas.
CLASES DE CONTROL:	
CLASES DE ENTIDAD:	Entidad de Tareas
FIGURA:	Figura 2.23
FLUJO DE SUCESOS	
1.- El usuario busca una tarea, evento o cita.	
2.- Se devuelve datos de tareas.	
3.- Se selecciona la tarea, evento o cita.	
4.- Se devuelve los datos del registro.	



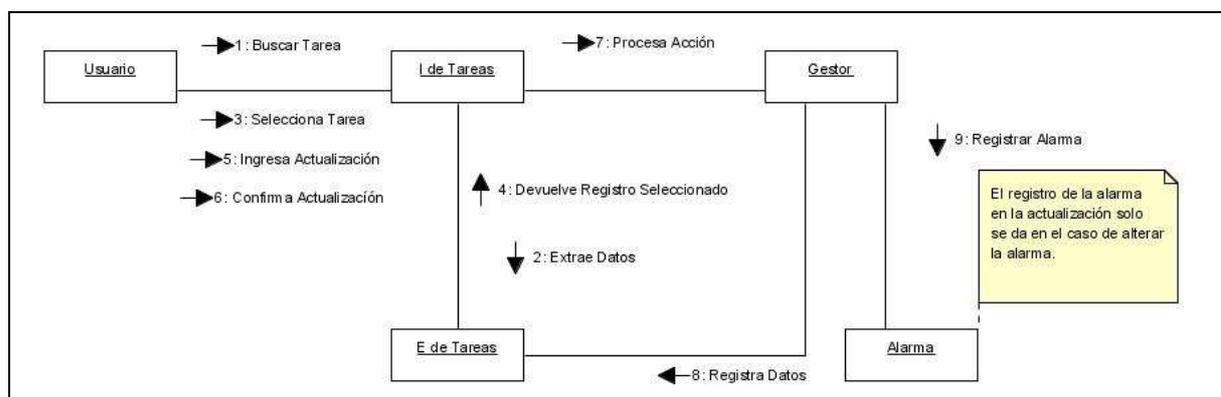
[Figura 2.23]

MODELO:	Modelo de Análisis
DIAGRAMA:	Diagrama de Colaboración
CASO DE USO:	Gestión de Tareas, Eventos o Citas – Eliminar tarea, evento o cita.
CLASES DE INTERFAZ:	Interfaz de Tareas, Eventos o Citas.
CLASES DE CONTROL:	Gestor
CLASES DE ENTIDAD:	Entidad de Tareas
FIGURA:	Figura 2.24
FLUJO DE SUCESOS	
<ol style="list-style-type: none"> 1.- El usuario busca una tarea, evento o cita. 2.- Se devuelve datos de tareas. 3.- Se selecciona la tarea, evento o cita. 4.- Se devuelve los datos del registro. 5.- El usuario confirma la eliminación. 6.- Se envía la orden de eliminación de la tarea hacia el gestor. 7.- Se actualiza el Registro. 	



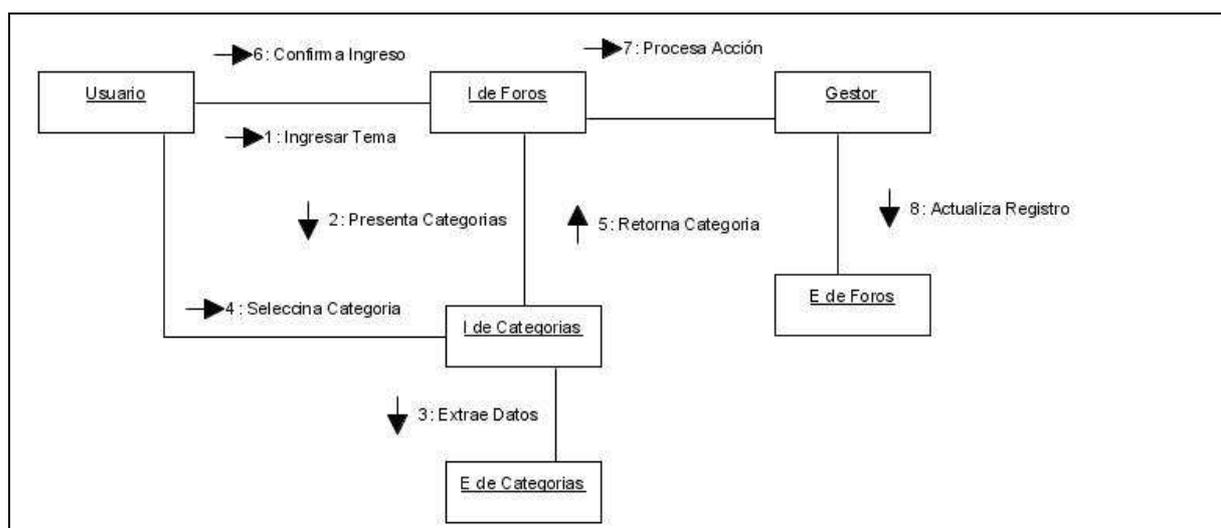
[Figura 2.24]

MODELO:	Modelo de Análisis
DIAGRAMA:	Diagrama de Colaboración
CASO DE USO:	Gestión de Tareas, Eventos o Citas – Actualizar tarea, evento o cita.
CLASES DE INTERFAZ:	Interfaz de Tareas, Eventos o Citas.
CLASES DE CONTROL:	Gestor
CLASES DE ENTIDAD:	Entidad de Tareas
FIGURA:	Figura 2.25
FLUJO DE SUCECOS	
<ol style="list-style-type: none"> 1.- El usuario busca una tarea, evento o cita. 2.- Se devuelve datos de tareas. 3.- Se selecciona la tarea, evento o cita. 4.- Se devuelve los datos del registro. 5.- El usuario ingresa la actualización. 6.- El usuario confirma la actualización. 7.- Se envía la orden de actualización de la tarea hacia el gestor. 8.- Se actualiza el Registro. 9.- Se actualiza la Alarma. (Opcional) 	



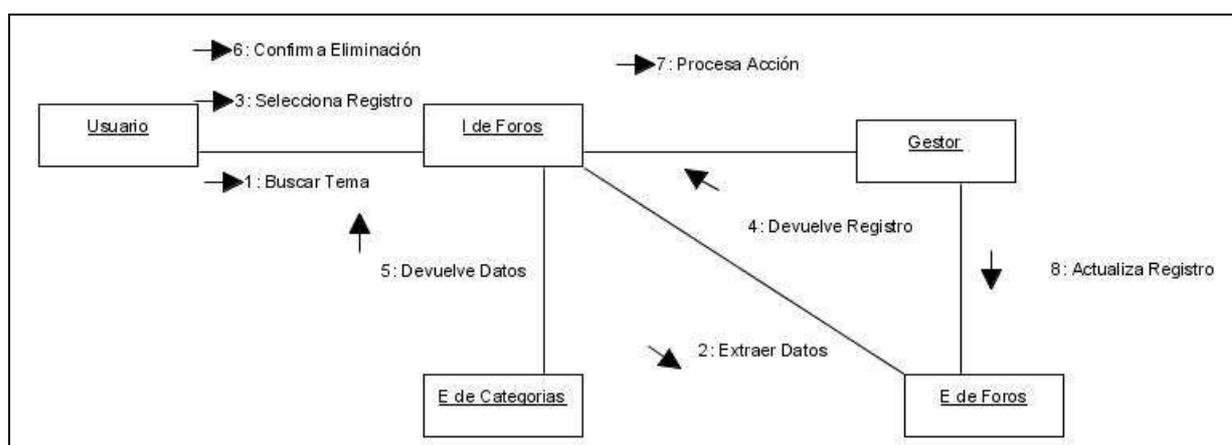
[Figura 2.25]

MODELO:	Modelo de Análisis
DIAGRAMA:	Diagrama de Colaboración
CASO DE USO:	Gestión de Foros de Discusión – Ingresar Tema a foro.
CLASES DE INTERFAZ:	Interfaz de Foros
	Interfaz de Categorías
CLASES DE CONTROL:	Gestor
CLASES DE ENTIDAD:	Entidad de Foros
	Entidad de Categorías
FIGURA:	Figura 2.26
FLUJO DE SUCESOS	
<ol style="list-style-type: none"> 1.- El usuario ingresa un tema con su respectiva información. 2.- Se presentan las categorías. 3.- Se extrae los datos. 4.- Se selecciona la categoría. 5.- Se devuelve información del registro seleccionado. 6.- El usuario confirma el ingreso del tema. 7.- Se envía la orden de ingreso del tema hacia el gestor. 8.- Se actualiza el registro. 	



[Figura 2.26]

MODELO:	Modelo de Análisis
DIAGRAMA:	Diagrama de Colaboración
CASO DE USO:	Gestión de Foros de Discusión – Eliminar Tema de foro.
CLASES DE INTERFAZ:	Interfaz de Foros
CLASES DE CONTROL:	Gestor
CLASES DE ENTIDAD:	Entidad de Foros
	Entidad de Categorías
FIGURA:	Figura 2.27
FLUJO DE SUCESOS	
<ol style="list-style-type: none"> 1.- El usuario decide consultar los temas disponibles. 2.- Se presentan los temas. 3.- Se selecciona el tema. 4.- Se devuelve información del registro seleccionado. 5.- Se devuelve datos. 6.- El usuario confirma la eliminación del tema. 7.- Se envía la orden de eliminación del tema hacia el gestor. 8.- Se actualiza el registro. 	



[Figura 2.27]

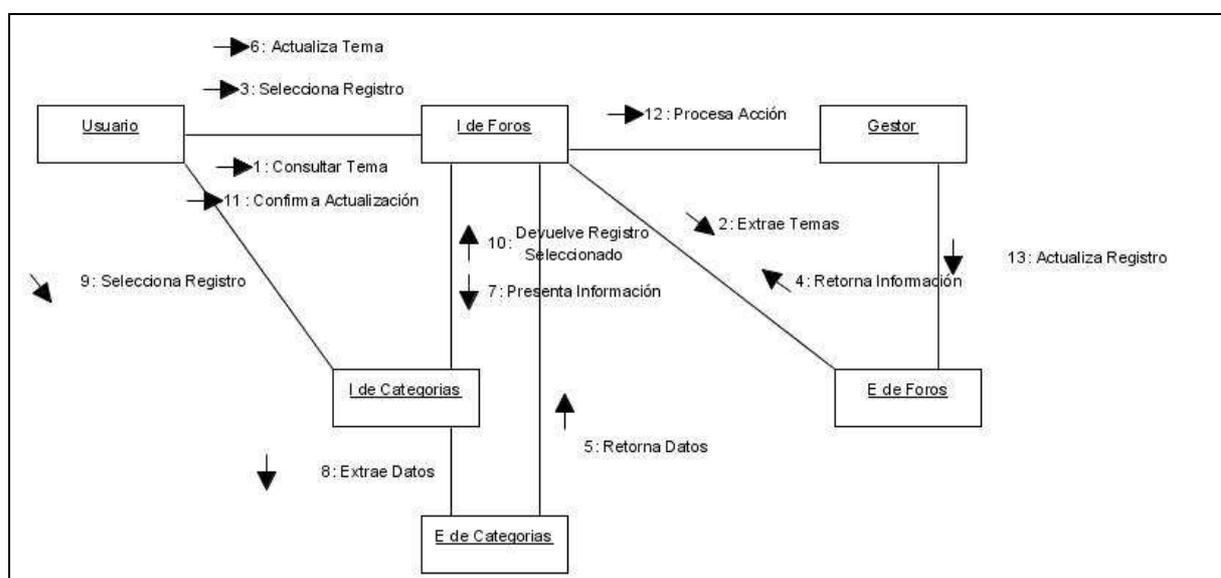
MODELO:	Modelo de Análisis
DIAGRAMA:	Diagrama de Colaboración
CASO DE USO:	Gestión de Foros de Discusión – Consultar tema en foro.
CLASES DE INTERFAZ:	Interfaz de Foros
CLASES DE CONTROL:	
CLASES DE ENTIDAD:	Entidad de Foros
FIGURA:	Figura 2.28
FLUJO DE SUCECOS	
<ol style="list-style-type: none"> 1.- El usuario decide consultar los temas disponibles. 2.- Se presentan los temas. 3.- Se selecciona el tema. 4.- Se devuelve información del registro seleccionado. 5.- Se retorna un listado con las respuestas al tema. 	



[Figura 2.28]

MODELO:	Modelo de Análisis
DIAGRAMA:	Diagrama de Colaboración
CASO DE USO:	Gestión de Foros de Discusión – Actualizar tema en foro.
CLASES DE INTERFAZ:	Interfaz de Foros
	Interfaz de Categorías
CLASES DE CONTROL:	Gestor
CLASES DE ENTIDAD:	Entidad de Foros

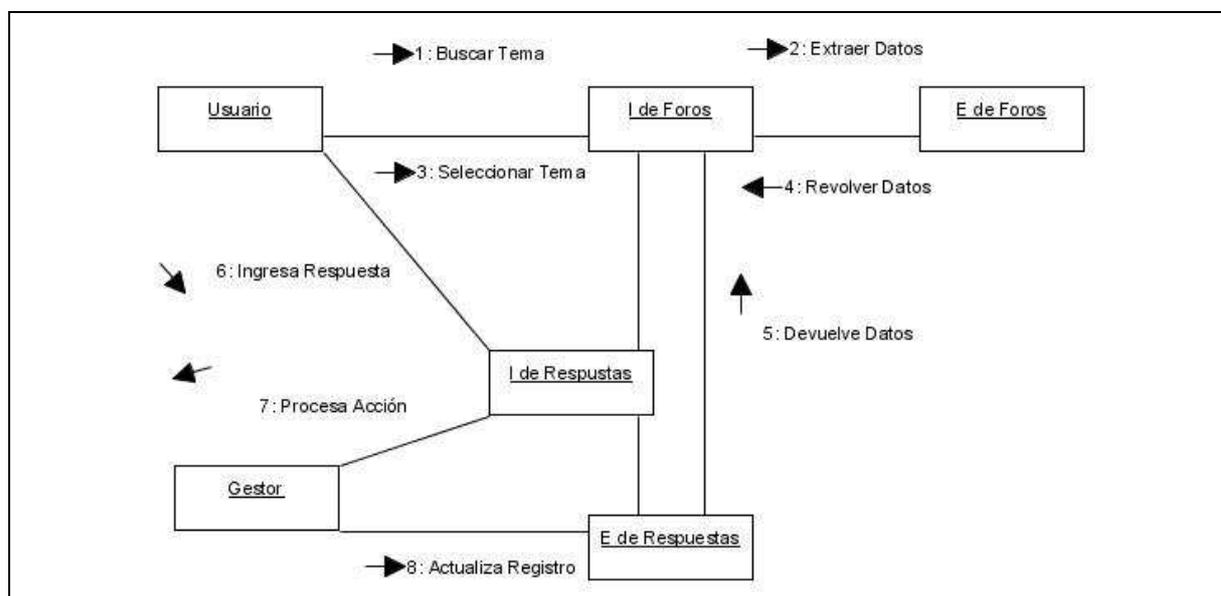
	Entidad de Categorías
FIGURA:	Figura 2.29
FLUJO DE SUCEOS	
<ol style="list-style-type: none"> 1.- El usuario decide consultar los temas disponibles. 2.- Se presentan los temas. 3.- El usuario selecciona el tema. 4.- Se devuelve información del registro seleccionado. 5.- Se retorna datos de categoría. 6.- El usuario actualiza la información del tema. 7.- Se presenta la información de las categorías. 8.- Se extrae los datos. 9.- Se selecciona el registro. 10.- Se devuelve el registro seleccionado. 11.- Se confirma la actualización. 12.- Se envía la orden de actualización del tema hacia el gestor. 13.- Se actualiza el registro. 	



[Figura 2.29]

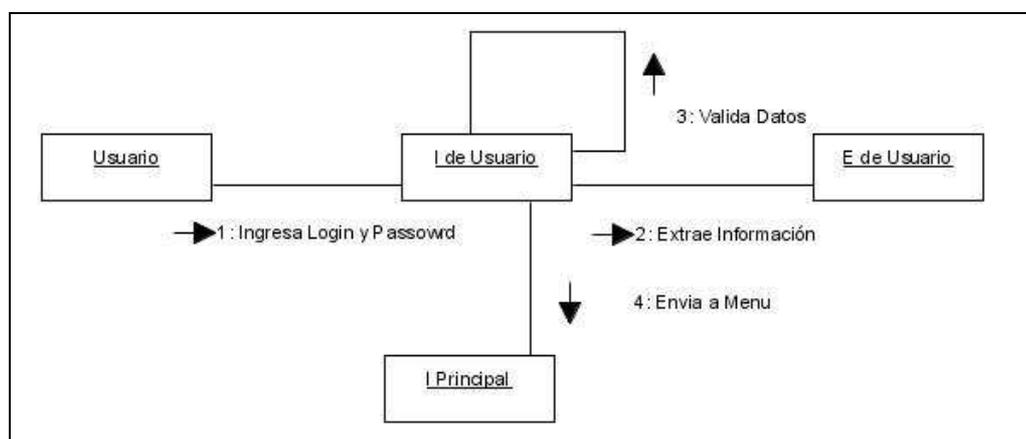
MODELO:	Modelo de Análisis
DIAGRAMA:	Diagrama de Colaboración
CASO DE USO:	Gestión de Foros de Discusión – Responder a un tema del foro.

CLASES DE INTERFAZ:	Interfaz de Foros
	Interfaz de Respuestas
CLASES DE CONTROL:	Gestor
CLASES DE ENTIDAD:	Entidad de Foros
	Entidad de Respuestas
FIGURA:	Figura 2.30
FLUJO DE SUCESOS	
<ol style="list-style-type: none"> 1.- El usuario decide consultar los temas disponibles. 2.- Se presentan los temas. 3.- El usuario selecciona el tema. 4.- Se devuelve información del registro seleccionado. 5.- Se retorna datos de las respuestas existentes. 6.- El usuario ingresa una respuesta. 7.- Se envía la orden de actualización de las respuestas del tema hacia el gestor. 8.- Se actualiza el registro. 	



[Figura 2.31]

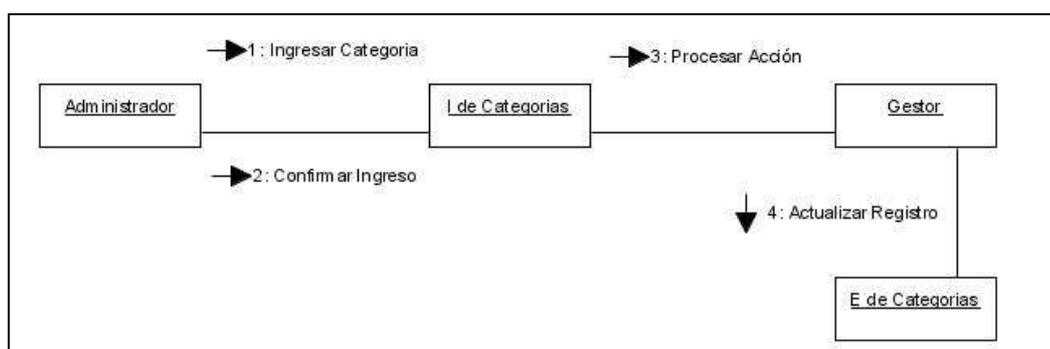
MODELO:	Modelo de Análisis
DIAGRAMA:	Diagrama de Colaboración
CASO DE USO:	Ingreso de usuario - Login
CLASES DE INTERFAZ:	Interfaz de Usuario
	Interfaz Principal
CLASES DE CONTROL:	
CLASES DE ENTIDAD:	Entidad de Foros
	Entidad de Respuestas
FIGURA:	Figura 2.32
FLUJO DE SUCESOS	
<ol style="list-style-type: none"> 1.- El usuario decide ingresar al sistema ingresando su nombre de usuario y contraseña. 2.- Se extrae la información para validar el ingreso. 3.- Se validan los datos. 4.- Se envía a la interfase principal del sistema. 	



[Figura 2.33]

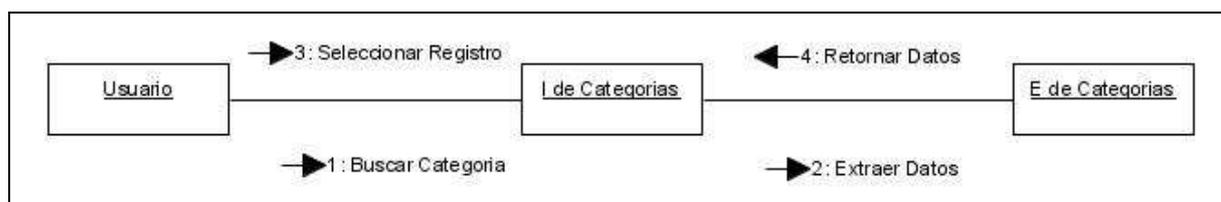
MODELO:	Modelo de Análisis
DIAGRAMA:	Diagrama de Colaboración
CASO DE USO:	Gestión de Categorías para foro – Ingresar Categoría de temas a foro.
CLASES DE INTERFAZ:	Interfaz de Categorías
CLASES DE CONTROL:	Gestor

CLASES DE ENTIDAD:	Entidad de Categorías
FIGURA:	Figura 2.34
FLUJO DE SUCECOS	
<ol style="list-style-type: none"> 1.- El usuario ingresa una categoría para un tema con su respectiva información. 2.- El usuario confirma el ingreso de la categoría. 3.- Se envía la orden de ingreso de la categoría hacia el gestor. 4.- Se actualiza el registro. 	



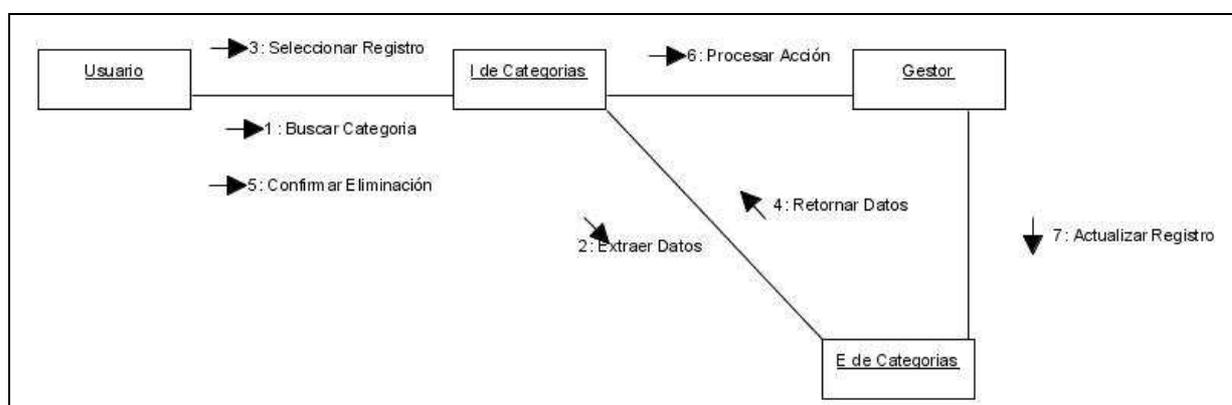
[Figura 2.34]

MODELO:	Modelo de Análisis
DIAGRAMA:	Diagrama de Colaboración
CASO DE USO:	Gestión de Categorías para foro – Consultar Categoría de temas para el foro.
CLASES DE INTERFAZ:	Interfaz de Categorías
CLASES DE CONTROL:	Gestor
CLASES DE ENTIDAD:	Entidad de Categorías
FIGURA:	Figura 2.35
FLUJO DE SUCECOS	
<ol style="list-style-type: none"> 1.- El usuario busca una categoría. 2.- Se extraen los datos de las categorías. 3.- El usuario selecciona un registro. 4.- Se devuelve la información del registro seleccionado. 	



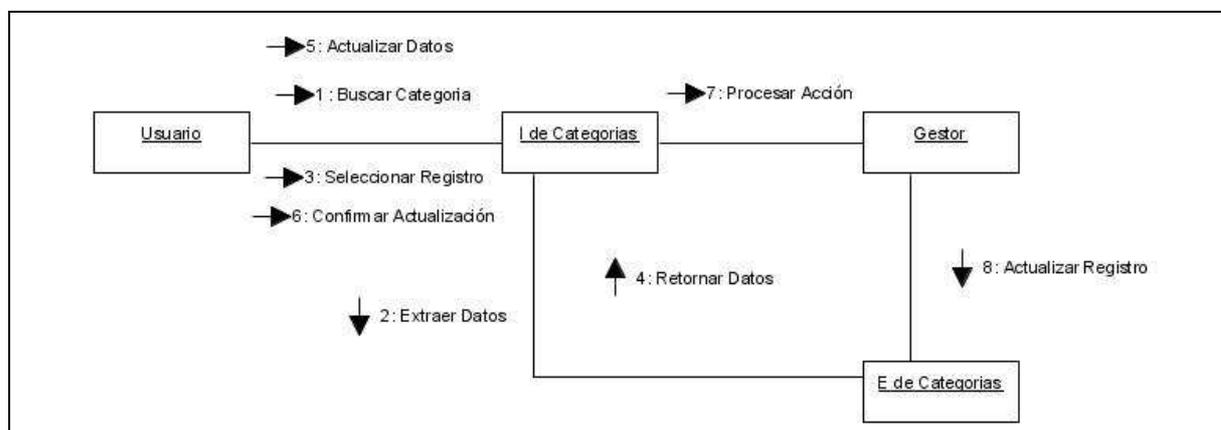
[Figura 2.35]

MODELO:	Modelo de Análisis
DIAGRAMA:	Diagrama de Colaboración
CASO DE USO:	Gestión de Categorías para foro – Eliminar Categoría del foro.
CLASES DE INTERFAZ:	Interfaz de Categorías
CLASES DE CONTROL:	Gestor
CLASES DE ENTIDAD:	Entidad de Categorías
FIGURA:	Figura 2.36
FLUJO DE SUCESOS	
<p>1.- El usuario busca una categoría.</p> <p>2.- Se extraen los datos de las categorías.</p> <p>3.- El usuario selecciona un registro.</p> <p>4.- Se devuelve la información del registro seleccionado.</p> <p>5.- El usuario confirma la eliminación.</p> <p>6.- Se envía la orden de ingreso de la categoría hacia el gestor.</p> <p>7.- Se actualiza la eliminación en el registro.</p>	



[Figura 2.36]

MODELO:	Modelo de Análisis
DIAGRAMA:	Diagrama de Colaboración
CASO DE USO:	Gestión de Categorías para foro – Actualizar Categoría del foro.
CLASES DE INTERFAZ:	Interfaz de Categorías
CLASES DE CONTROL:	Gestor
CLASES DE ENTIDAD:	Entidad de Categorías
FIGURA:	Figura 2.37
FLUJO DE SUCECOS	
<ol style="list-style-type: none"> 1.- El usuario busca una categoría. 2.- Se extraen los datos de las categorías. 3.- El usuario selecciona un registro. 4.- Se devuelve la información del registro seleccionado. 5.- El usuario ingresa los datos para la actualización. 6.- El usuario confirma la actualización. 7.- Se envía la orden de actualización de la categoría hacia el gestor. 8.- Se actualiza el registro. 	



[Figura 2.37]

2.2.1.4.- Identificación de clases, responsabilidades y atributos

Gestión de Usuario

Clase	Interfaz de Usuario	
Responsabilidad	<p>Su responsabilidad es la de receptor de información.</p> <p>Implica:</p> <ul style="list-style-type: none"> • Solicitar información sobre perfiles • Determina el inicio de la administración del usuario, en lo cual se efectuara el ingreso, actualización y eliminación del usuario. 	
Tipo	Interfaz	
Atributos Conceptuales	Nombre	Tipo
	Nombres	Texto
	Apellidos	Texto
	Dirección 1	Texto
	Dirección 2	Texto
	Teléfono 1	Texto
	Teléfono 2	Texto
	Perfil	Texto
	Contraseña	Texto
	Usuario	Texto
	Cargo	Texto
	Email	Texto
	Cédula	Texto

Clase	Entidad de Usuario	
Responsabilidad	<p>Su responsabilidad es la de almacenar la información del usuario.</p> <p>Implica:</p> <ul style="list-style-type: none"> • Devolver información sobre el usuario. • Aceptar ingresos, cambios y eliminaciones. 	
Tipo	Entidad	
Atributos Conceptuales	Nombre	Tipo

	Nombres	Texto
	Apellidos	Texto
	Dirección 1	Texto
	Dirección 2	Texto
	Teléfono 1	Texto
	Teléfono 2	Texto
	Código Perfil	Numérico
	Contraseña	Texto
	Usuario	Texto
	Cargo	Texto
	Email	Texto
	Cédula o Identificación	Numérico
	Código Interno	Numérico

Clase	Gestor de Usuario
Responsabilidad	Su responsabilidad es la de recibir información para ingresar, actualizar o eliminar la información de un registro. Implica: <ul style="list-style-type: none"> • Ingresar y actualizar datos del nuevo usuario. • Eliminar registro.
Tipo	Control
Atributos Conceptuales	Variables temporales

Clase	Interfaz de Consulta de Usuario	
Responsabilidad	Su responsabilidad es la de recibir información. Implica: <ul style="list-style-type: none"> • Solicitar información sobre perfiles. • Despliega información de los usuarios. 	
Tipo	Interfaz	
Atributos Conceptuales	Nombre	Tipo
	Nombres	Texto
	Apellidos	Texto

	Dirección 1	Texto
	Dirección 2	Texto
	Teléfono 1	Texto
	Teléfono 2	Texto
	Perfil	Texto
	Cargo	Texto
	Email	Texto
	Cédula	Texto

Gestión de Perfiles

Clase	Interfaz de Perfil	
Responsabilidad	<p>Su responsabilidad es la de aceptar información.</p> <p>Implica:</p> <ul style="list-style-type: none"> • Determina el inicio de la administración del perfil, en lo cual se efectuara el ingreso, actualización y eliminación de un perfil. • Solicitar información de las funciones del sistema existentes o registradas. 	
Tipo	Interfaz	
Atributos	Nombre	Tipo
	Nombre	Texto
	Descripción	Texto

Clase	Interfaz de Consulta de Perfil	
Responsabilidad	<p>Su responsabilidad es la mostrar información.</p> <p>Implica:</p> <ul style="list-style-type: none"> • Mostrar información sobre perfiles. • Devolver perfil seleccionado 	
Tipo	Interfaz	
Atributos	Nombre	Tipo
	Nombre	Texto
	Descripción	Texto

Clase	Entidad de Perfil	
Responsabilidad	Su responsabilidad es la de almacenar el ingreso de la información. Implica: <ul style="list-style-type: none"> • Devolver información sobre el perfil. • Aceptar ingresos, cambios y eliminaciones. 	
Tipo	Entidad	
Atributos	Nombre	Tipo
	Nombres	Texto
	Descripción	Texto
	Código Interno	Numérico

Clase	Gestor de Perfiles	
Responsabilidad	Su responsabilidad es la de receptor información para ingresar, actualizar o eliminar la información de un registro. Implica: <ul style="list-style-type: none"> • Ingresar y actualizar datos del nuevo perfil. • Eliminar registro. 	
Tipo	Control	
Atributos	Variables temporales	

Gestión de la Organización

Clase	Interfaz de la Organización	
Responsabilidad	Su responsabilidad es la de receptor información. Implica: <ul style="list-style-type: none"> • Determina el inicio de la administración de los datos, en lo cual se efectuara el ingreso y la actualización. 	
Tipo	Interfaz	
Atributos	Nombre	Tipo
	Nombre	Texto
	Dirección	Texto
	Descripción	Texto

	Representante	Texto
Clase	Entidad de la Organización	
Responsabilidad	<ul style="list-style-type: none"> Su responsabilidad es la de almacenar la información de la organización. 	
Tipo	Entidad	
Atributos	Nombre	Tipo
	Nombre	Texto
	Dirección	Texto
	Descripción	Texto
	Representante	Texto

Clase	Gestor de la Organización	
Responsabilidad	<p>Su responsabilidad es la de recibir información para ingresar, actualizar la información de los datos de la organización.</p> <p>Implica:</p> <ul style="list-style-type: none"> Ingresar y actualizar datos de la organización. 	
Tipo	Control	
Atributos	Variables temporales	

Gestión de Funciones

Clase	Interfaz de consulta de Funciones	
Responsabilidad	<p>Su responsabilidad es la de mostrar información.</p> <p>Implica:</p> <ul style="list-style-type: none"> Mostrar listado de funciones disponibles Retornar funciones escogidas. 	
Tipo	Interfaz	
Atributos	Nombre	Tipo
	Nombre	Texto
	Descripción	Texto
	Dirección de la página	Texto
	Nombre	Texto

Clase	Interfaz de Funciones	
Responsabilidad	<p>Su responsabilidad es la de recibir información.</p> <p>Implica:</p> <ul style="list-style-type: none"> • Determina el inicio de la administración de las funciones del sistema, en lo cual se efectuará el ingreso, actualización y eliminación de una función. 	
Tipo	Interfaz	
Atributos	Nombre	Tipo
	Nombre	Texto
	Descripción	Texto
	Path Página	Texto
	Nombre	Texto

Clase	Entidad de Funciones	
Responsabilidad	<p>Su responsabilidad es la de proporcionar información del registro.</p> <p>Implica:</p> <ul style="list-style-type: none"> • Devolver información sobre la función. • Aceptar ingresos, cambios y eliminaciones. 	
Tipo	Entidad	
Atributos	Nombre	Tipo
	Nombre	Texto
	Descripción	Texto
	Path Página	Texto
	Código	Número
	Nombre	Texto

Clase	Gestor de Funciones	
Responsabilidad	<p>Su responsabilidad es la de recibir información para ingresar, actualizar o eliminar la información de un registro.</p> <p>Implica:</p>	

	<ul style="list-style-type: none"> • Ingresar y actualizar datos de funciones y eliminarlos.
Tipo	Control
Atributos	Variables temporales

Gestión de Grupos de Trabajo

Clase	Interfaz de consulta de Grupos de Trabajo	
Responsabilidad	Su responsabilidad es la de receptor de información. Implica: <ul style="list-style-type: none"> • Solicitar datos de búsqueda de grupos. • Mostrar listado de grupos disponibles • Retornar grupo escogido 	
Tipo	Interfaz	
Atributos	Nombre	Tipo
	Nombre	Texto
	Descripción	Texto
	Búsqueda	Texto

Clase	Interfaz de Grupos de Trabajo	
Responsabilidad	Su responsabilidad es la de receptor de información. Implica: <ul style="list-style-type: none"> • Determina el inicio de la administración de los grupos de trabajo disponibles en el sistema, y en la cual se efectuara el ingreso, actualización y eliminación de un grupo. 	
Tipo	Interfaz	
Atributos	Nombre	Tipo
	Nombre	Texto
	Código Id.	Numérico
	Descripción	Texto
	Fecha de creación	Fecha

Clase	Entidad de Grupos de Trabajo	
Responsabilidad	<p>Su responsabilidad es la de proporcionar información del registro.</p> <p>Implica:</p> <ul style="list-style-type: none"> • Devolver información sobre los grupos de trabajo. • Aceptar ingresos, cambios y eliminaciones de los registros que guarda. 	
Tipo	Entidad	
Atributos	Nombre	Tipo
	Nombre	Texto
	Código Id.	Texto
	Descripción	Texto
	Fecha de creación	Numérico

Clase	Gestor de Grupos de Trabajo	
Responsabilidad	<p>Su responsabilidad es la de aceptar información para ingresar, actualizar o eliminar la información de un registro.</p> <p>Implica:</p> <ul style="list-style-type: none"> • Ingresar y actualizar datos de los grupos formados. • Eliminar registro/s. 	
Tipo	Control	
Atributos	Variables temporales	

Gestión de Tareas

Clase	Interfaz de consulta de Tareas	
Responsabilidad	<p>Su responsabilidad es la de aceptar información.</p> <p>Implica:</p> <ul style="list-style-type: none"> • Solicitar datos de búsqueda de tareas, eventos, citas, reuniones. • Mostrar listado de tareas. • Retornar tarea escogida. 	
Tipo	Interfaz	

Atributos	Nombre	Tipo
	Tema	Texto
	Descripción	Texto
	Tipo	Texto
	Estado	Texto
	Prioridad	Texto
	Avance	Texto
	Fecha Inicio	Texto
	Fecha Fin	Texto
	Fecha Creación	Texto
	Hora Inicio	Texto
	Hora Fin	Texto
	Búsqueda	Texto

Clase	Interfaz de consulta de Tareas	
Responsabilidad	<p>Su responsabilidad es la de recibir información. Implica:</p> <ul style="list-style-type: none"> Determina el inicio de la administración de las tareas con su correspondiente tipo y que se encuentren disponibles en el sistema, y en la cual se efectuara el ingreso, actualización y eliminación de una tarea. 	
Tipo	Interfaz	
Atributos	Nombre	Tipo
	Tema	Texto
	Descripción	Texto
	Tipo	Texto
	Estado	Texto
	Prioridad	Texto
	Avance	Texto
	Fecha Inicio	Texto
	Fecha Fin	Texto
	Fecha Creación	Texto
	Hora Inicio	Texto

	Hora Fin	Texto
--	----------	-------

Clase	Entidad de Tareas	
Responsabilidad	Su responsabilidad es la de proporcionar información del o los registro/s. Implica: <ul style="list-style-type: none"> • Devolver información sobre las tareas. • Aceptar ingresos, cambios y eliminaciones. 	
Tipo	Entidad	
Atributos	Nombre	Tipo
	ID	Numérico
	Tema	Texto
	Descripción	Texto
	Tipo	Texto
	Estado	Texto
	Prioridad	Texto
	Avance	Texto
	Fecha Inicio	Fecha
	Fecha Fin	Fecha
	Fecha Creación	Fecha
	Hora Inicio	Texto
	Hora Fin	Texto

Clase	Gestor de Tareas	
Responsabilidad	Su responsabilidad es la de recibir información para ingresar, actualizar o eliminar la información de un registro. Implica: <ul style="list-style-type: none"> • Ingresar y actualizar datos de las tareas del usuario. • Eliminar registro/s. 	
Tipo	Control	
Atributos	Variables temporales	

Gestión de Foros de Discusión

Clase	Interfaz de consulta de Foros de Discusión	
Responsabilidad	Su responsabilidad es la de recibir información. Implica: <ul style="list-style-type: none"> • Solicitar datos de búsqueda de temas en el foro. • Mostrar listado de temas. • Retornar el tema escogido. 	
Tipo	Interfaz	
Atributos	Nombre	Tipo
	Tema	Texto
	Categoría	Texto
	Descripción	Texto
	Fecha de Publicación	Texto
	Autor del Tema	Texto
	Búsqueda	Texto

Clase	Interfaz de Foros de Discusión	
Responsabilidad	Su responsabilidad es la de recibir información. Implica: <ul style="list-style-type: none"> • Determina el inicio de la administración de los temas del foro de discusiones con sus correspondientes respuestas y que se encuentren disponibles en el sistema, y en la cual se efectuara el ingreso, actualización y eliminación de un tema. 	
Tipo	Interfaz	
Atributos	Nombre	Tipo
	Tema	Texto
	Categoría	Texto
	Descripción	Texto
	Fecha de Publicación	Texto
	Autor del Tema	Texto

Clase	Interfaz de Respuestas a Temas del Foro	
Responsabilidad	<p>Su responsabilidad es la de recibir información. Implica:</p> <ul style="list-style-type: none"> • Solicitar datos de búsqueda de repuestas a temas en el foro. • Mostrar listado de repuestas a temas. • Retornar repuestas de tema escogido. 	
Tipo	Interfaz	
Atributos	Nombre	Tipo
	Tema	Texto
	Categoría	Texto
	Descripción	Texto
	Fecha de Publicación	Texto
	Autor del Tema	Texto
	Autor de la Respuesta	Texto
	Fecha de Respuesta	Texto
	Listado de Respuestas	Texto
	Búsqueda	Texto

Clase	Entidad de Respuestas a Temas del Foro	
Responsabilidad	<p>Su responsabilidad es la de proporcionar información del registro. Implica:</p> <ul style="list-style-type: none"> • Devolver información sobre las respuestas del tema. • Aceptar ingresos, cambios y eliminaciones de los registros que guarda. 	
Tipo	Entidad	
Atributos	Nombre	Tipo
	ID Tema	Numérico
	Respuesta	Texto
	Fecha Respuesta	Texto
	ID de Autor de Respuesta	Numérico
	ID de Respuesta	Numérico

Clase	Entidad de Foros de Discusión	
Responsabilidad	Su responsabilidad es la de proporcionar información del registro. Implica: <ul style="list-style-type: none"> • Devolver información sobre el tema. • Aceptar ingresos, cambios y eliminaciones de los registros que guarda. 	
Tipo	Entidad	
Atributos	Nombre	Tipo
	Tema	Texto
	ID Categoría	Numérico
	Descripción	Texto
	Fecha de Publicación	Fecha
	ID Autor del Tema	Numérico

Clase	Gestor de Foros de Discusión	
Responsabilidad	Su responsabilidad es la de recibir información para ingresar, actualizar o eliminar la información de un registro en la entidad el tema y en la entidad de respuestas Implica: <ul style="list-style-type: none"> • Ingresar y actualizar datos de los temas. • Ingresar y actualizar datos de las respuestas de los temas. • Eliminar registro/s. 	
Tipo	Control	
Atributos	Variables temporales	

Ingreso

Clase	Interfaz de Inicio de Sesión
Responsabilidad	Su responsabilidad es la de recibir información. Implica:

	<ul style="list-style-type: none"> • Solicitar datos para validar el ingreso al sistema. • Enviar a menú principal. 	
Tipo	Interfaz	
Atributos	Nombre	Tipo
	Usuario	Texto
	Contraseña	Texto

Gestión de Categorías

Clase	Interfaz de consulta de Categorías	
Responsabilidad	Su responsabilidad es la de receptor de información. Implica: <ul style="list-style-type: none"> • Solicitar datos de búsqueda de categorías. • Mostrar listado de categorías del foro. • Retornar la categoría escogida. 	
Tipo	Interfaz	
Atributos	Nombre	Tipo
	Nombre	Texto
	Descripción	Texto
	Activo	Texto
	Búsqueda	Texto

Clase	Interfaz de consulta de Categorías	
Responsabilidad	Su responsabilidad es la de receptor de información. Implica: <ul style="list-style-type: none"> • Determina el inicio de la administración de las categorías del foro de discusión y que se encuentren disponibles en el sistema, y en la cual se efectuara el ingreso, actualización y eliminación de una categoría. 	
Tipo	Interfaz	
Atributos	Nombre	Tipo
	Nombre	Texto
	Activo	Texto
	Descripción	Texto

Clase	Entidad de Categorías	
Responsabilidad	Su responsabilidad es la de proporcionar información del registro. Implica: <ul style="list-style-type: none"> • Devolver información sobre la categoría. • Aceptar ingresos, cambios y eliminaciones de los registros que guarda. 	
Tipo	Entidad	
Atributos	Nombre	Tipo
	Nombre	Texto
	Descripción	Texto
	ID de Categoría	Numérico
	Activo	Numérico

Clase	Gestor de Categorías del Foro	
Responsabilidad	Su responsabilidad es la de aceptar información para ingresar, actualizar o eliminar la información de un registro. Implica: <ul style="list-style-type: none"> • Ingresar y actualizar datos de las categorías. • Eliminar registro/s. 	
Tipo	Control	
Atributos	Variables temporales	

2.2.1.5.- Diagrama de Paquetes de la Aplicación

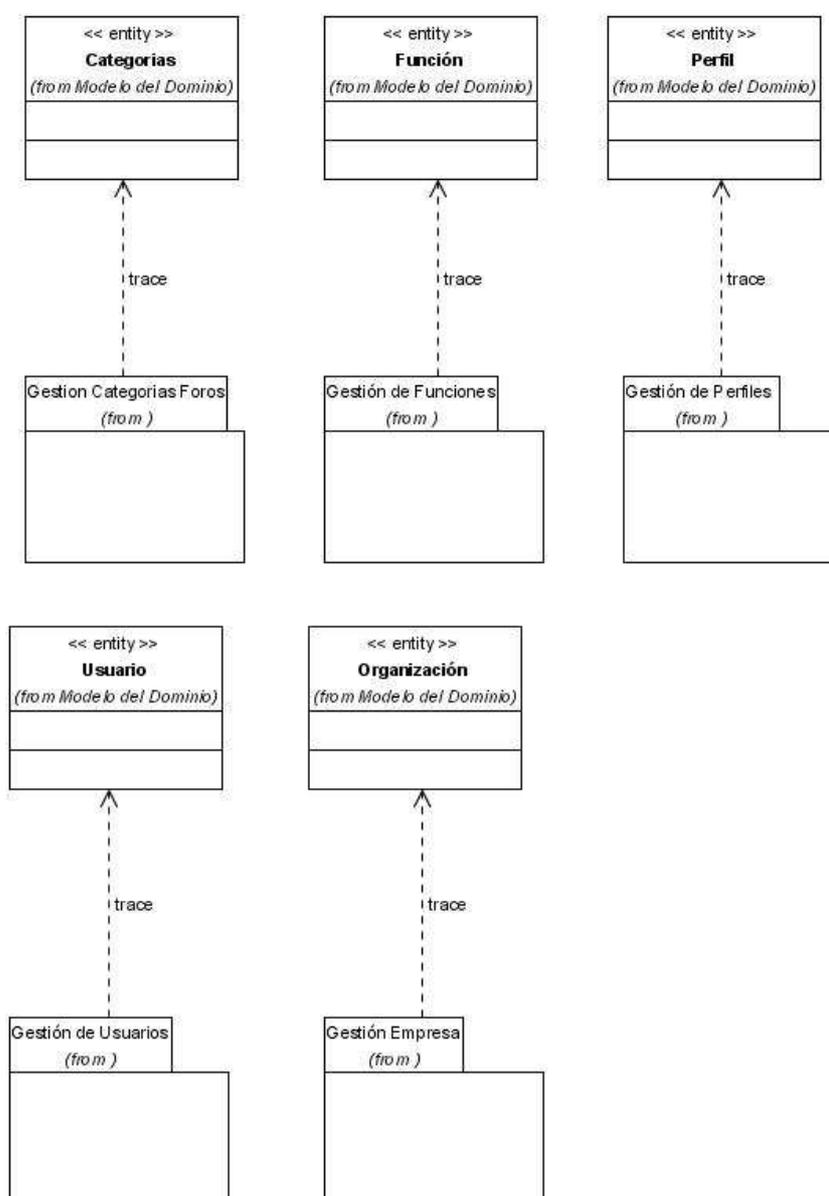
Los paquetes de análisis son artefactos que organizan las clases de análisis, reuniendo y agrupando clases que estén relacionadas entre sí, en paquetes que sean manejables, estos paquetes son representados mediante diagramas de paquetes.

Para esta aplicación identificaremos los paquetes de análisis recurriendo a las relaciones que existen entre las clases, los requisitos funcionales representados por los casos de uso y el conjunto de servicios que estos puedan tener con el

usuario Administrador [Figura 2.38] y con el usuario normal del Sistema [Figura 2.39].

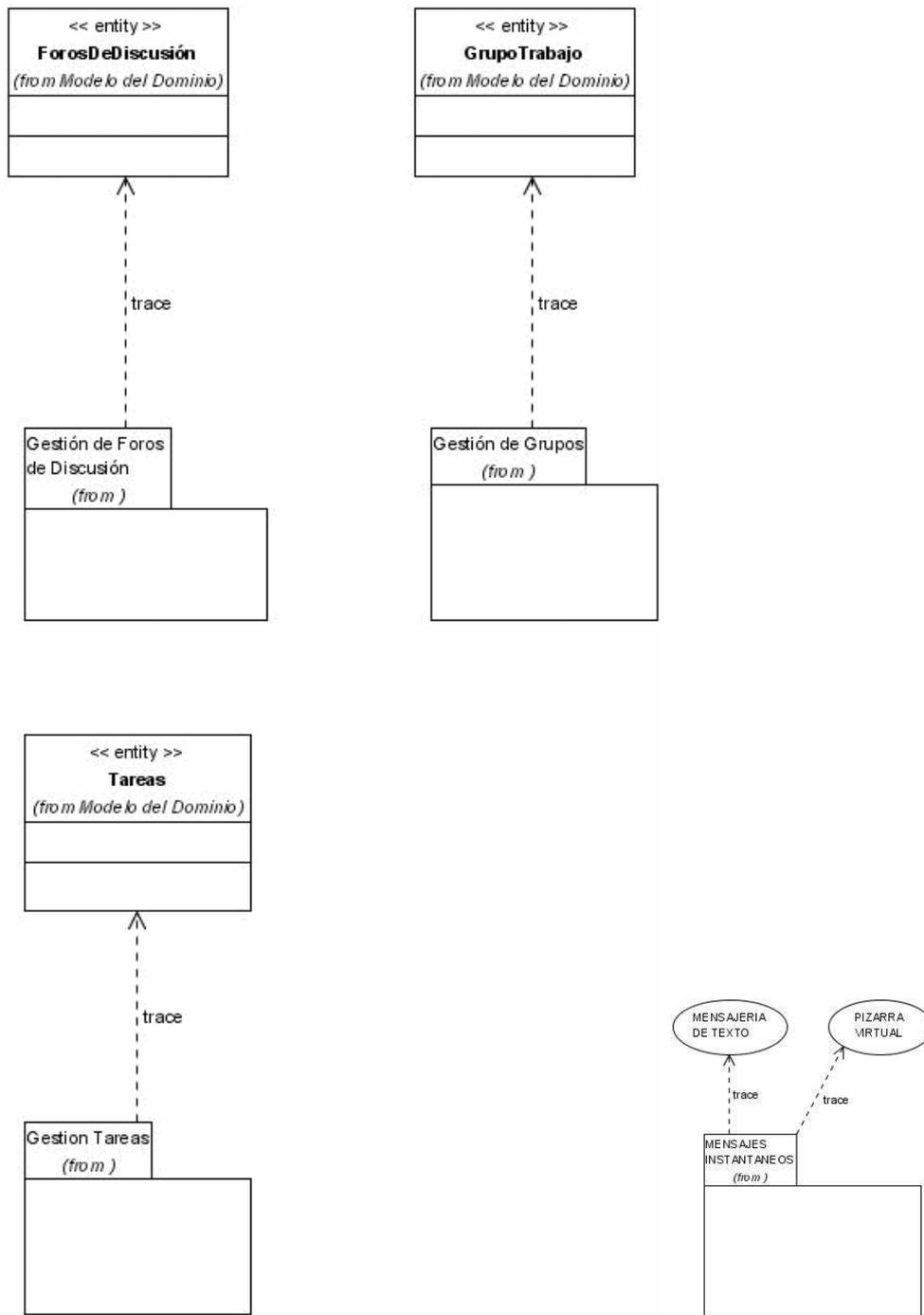
Como complemento al diagrama de paquetes se presentara el diagrama de dependencia de paquetes [Figura 2.40] el cual presentara a los paquetes con las dependencias que necesita para realizar sus funciones.

Aplicación Administrador



[Figura 2.38]

APLICACIÓN USUARIO

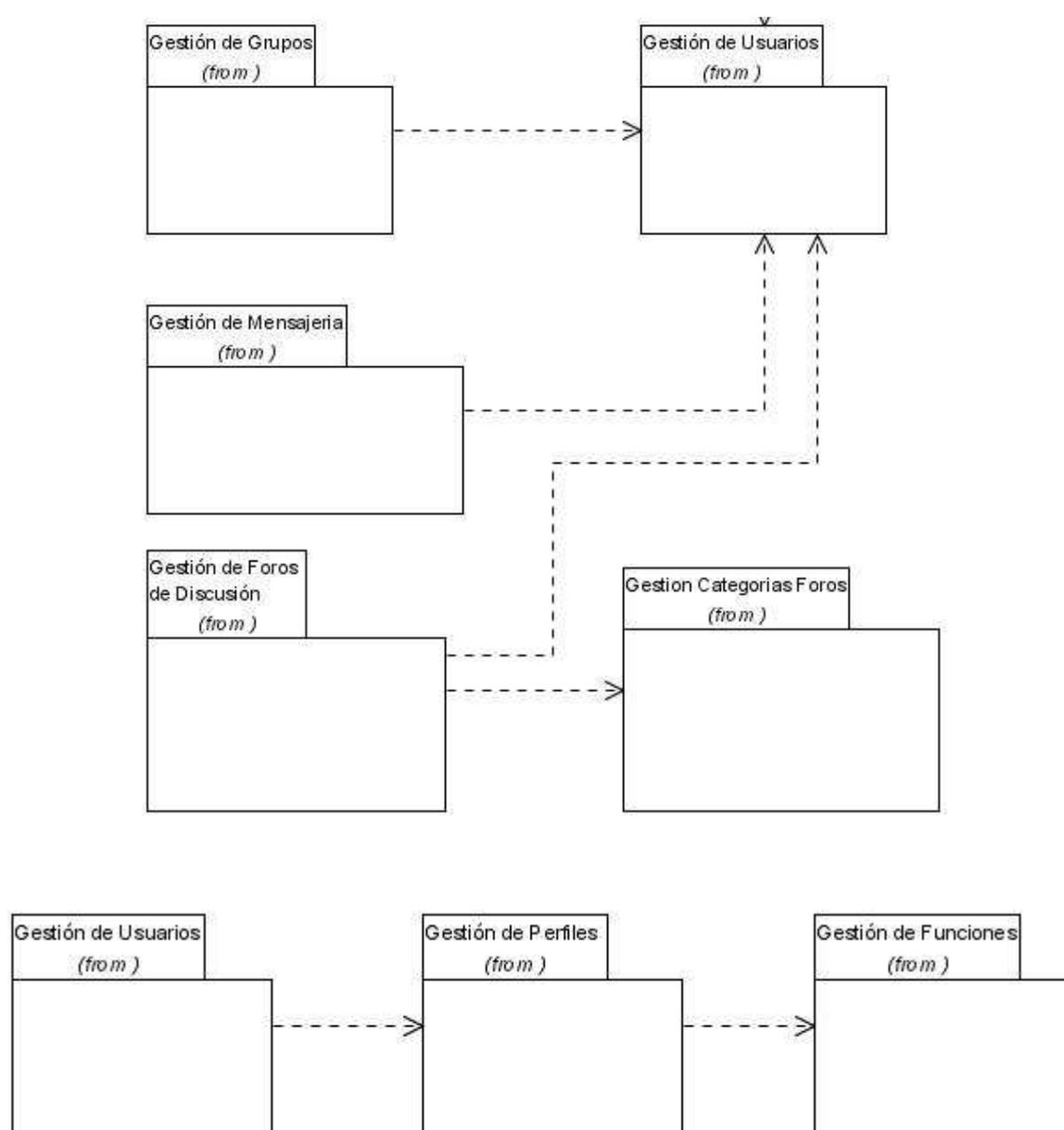


[Figura 2.39]

Elaborado por: El Autor

Referencia: Modelos desarrollados por el autor con la herramienta Poseidón.

2.2.1.6.- Dependencia entre Paquetes de la Aplicación



[Figura 2.40]

Elaborado por: El Autor

Referencia: Modelos desarrollados por el autor con la herramienta Poseidón.

2.3.- COMPARACIÓN DE TECNOLOGÍAS PARA EL USO DE MENSAJERA ELECTRÓNICA

La orientación a objetos exige que se la trabaje con una plataforma que brinde capacidades de desarrollo simple, distribuido, robusto y de carácter empresarial como son:

- Corba
- Punto Net
- J2EE

Corba es una plataforma empresarial que presenta ciertas características como son:

- El soporte de múltiples sistemas operativos, soporta múltiples lenguajes de desarrollo y provee servicios como son: Mensajería, Eventos, Transacciones, Persistencia.
- El desarrollo es complejo y por ende la creación aplicaciones, también es burocrático lo cual no le permite evolucionar y por esto Corba no mejora.
- El no evolucionar provoca que su desarrollo e implementaciones se estanquen.

Punto Net, es una plataforma cerrada ya que depende de Microsoft, pero es robusta ya que ofrece características como:

- En desarrollo, ya que Visual Studio Punto Net es un producto el cual engloba o encapsula toda la plataforma de desarrollo, para crear una aplicación la plataforma Punto Net tiene todo lo que se necesita para que funcione dicha aplicación.
- Soporta múltiples lenguajes de desarrollo, entre ellos tenemos a C#, Visual Basic, C++, entre otros. Pero lo que Punto Net muestra es una interoperatividad entre los lenguajes permitiendo que se pueda hacer uso de componentes creados en uno de los lenguajes y utilizarlo en otros componentes con diferentes lenguajes.
- En Rendimiento Visual Studio Punto Net es evaluado sobre el hardware empleado, por ende depende del equipo en el que este funcionando. Pero indistintamente del equipo, este debe ser sofisticado a razón de

componentes internos del equipo que permitan características de transmisión de datos con agilidad, y de procesamiento rápido para una compilación con agilidad.

- En Portabilidad Visual Studio Punto Net trabaja única y exclusivamente sobre el sistema operativo Windows ya que el framework se integra al sistema operativo.
- En Transaccionalidad los usuarios o desarrolladores que trabajen sobre Visual Studio Punto Net mantienen las propiedades ACID (Atomicidad, Consistencia, Aislamiento y Durabilidad), de dos maneras: Transacciones manuales que son programando el inicio de la transacción, la ejecución, si la abortan o la finalizan, y las transacciones automáticas asignando un valor o semáforo a un objeto y viendo su comportamiento.
- En Seguridad Visual Studio Punto Net trabaja con servicios proporcionados por el sistema operativo para la autenticación y autorización. Lo cual significa que esta ligado a la configuración de seguridad del sistema operativo.
- En Escalabilidad Visual Studio Punto Net permite realizar carga balanceada con lo cual puede dar un servicio en forma simultánea esto con un cluster de servidores.
- No se necesita desarrolladores experimentados para poder desarrollar, Visual Studio también provee servicios de Mensajería, Eventos, Transacciones, Servicios Web.
- También es probablemente la mejor solución en el caso de que si se dispone de gran cantidad de equipos que trabajan con productos Microsoft, es mejor continuar trabajando con productos que sean compatibles y que se integren sin mayor problema.
- El costo de implantación de Visual Studio Punto Net depende de su propietario en este caso Microsoft.

J2EE es una especificación que define una plataforma de tecnología de desarrollo empresarial, fue creada por SUN Microsystems y que basa su arquitectura en productos de software libre, cuyos componentes son: Un conjunto de especificaciones, un test de compatibilidad, la implantación de referencias de J2EE y un conjunto de guías de desarrollo.

Sus principales características son:

- En desarrollo al ser J2EE un estándar, existen algunos entornos desarrollados que cumplen con las especificaciones que el estándar requiere y con lo cual se abre a algunas posibilidades dependiendo del usuario programador a cual se acople mejor.
- El lenguaje de programación es JAVA siendo este el único lenguaje para el desarrollo de todos los componentes.
- En Rendimiento J2EE al ser una plataforma con más tiempo en el mercado con lo cual han madurado en los productos utilizados lo que brindan un rendimiento óptimo ya que se han corregido los errores, se han cubierto las necesidades y tiene empresas que han consolidado los productos para el desarrollo en J2EE. También depende del hardware sobre el cual trabaje y complemente los tiempos de respuesta como de procesamiento y compilación.
- En Portabilidad J2EE puede ser utilizada en cualquier sistema operativo porque trabaja con la maquina virtual de java la cual permite su ejecución.
- En Transaccionalidad J2EE permite a los usuarios programadores gestionar la forma de trabajar con las transacciones ya sea esta manual en la cual el programador se encarga de iniciar la transacción, la ejecución, si la abortan o la finalizan, o especificar el comportamiento requerido y configurar para que la gestión la realice el contener EJB con lo cual se evita errores por parte del programador.
- En Seguridad J2EE permite a los usuarios programadores realizar sus propios controles de autenticación y autorización para la ejecución de las clases, pero aunque no es requerido si existe una funcionalidad de autenticación y autorización que es proporcionada por JAAS (Java Authentication and Authorisation Service). También puede examinar la procedencia de las clases mediante el Class-Loader, con lo cual se puede hacer uso de certificados digitales para la ejecución de clases como los applets.
- En Escalabilidad J2EE soporta métodos para soportar mas carga de trabajo como por ejemplo balancear carga en la cual permite a un cluster de varios servidores dar servicio en forma simultanea.

- En costos de implantación de J2EE pueden ser o no económico esto dependiendo de la complejidad de la aplicación, las herramientas utilizadas sean gratuitas o no.
- J2EE tiene un organismo de control llamado JCP el cual esta conformado por mas de 500 empresas y cuyo objetivo es de asegurar la evolución de las plataformas basadas en JAVA.

MATRIZ COMPARATIVA DE CARACTERÍSTICAS EN TECNOLOGÍAS DE DESARROLLO

TECNOLOGÍA CARACTERÍSTICAS	J2EE	PUNTO NET	CORBA
Rendimiento(Hardware)	4	4	3
Portabilidad	5	1	3
Manejado por estándares	5	1	2
Transaccionalidad	3	3	1
Seguridad	4	4	3
Escalabilidad(carga de trabajo)	4	4	2
Costo	4	2	3
Versatilidad en el uso de la herramienta(s).	3	5	1
Orientación a objetos	5	5	2
Experiencia en el mercado.	5	4	3
No depende de otros productos.	3	5	1
Tiene servicios de: Mensajería, Eventos, Transacciones.	5	5	4
Permite evolucionar a la tecnología	5	5	2
Tecnología no propietaria.	5	1	3
Tiene un organismo que controla la tecnología.	5	1	1
Mantenimiento	4	3	2
Total	69	53	36

Muy Alto 5 Alto 4 Medio 3 Bajo 2 Muy Bajo 1

Hay que aclarar que el puntaje dado a cada característica está acorde al punto de vista del autor del presente proyecto y debe ser tomado únicamente como una referencia al igual que la sumatoria del puntaje.

Se escogió la plataforma J2EE por las prestaciones y ventajas que su lenguaje de desarrollo en JAVA tiene, el manejo orientado a objetos, los diferentes sistemas operativos en los cuales se puede trabajar y la funcionalidad de sus especificaciones que para el presente proyecto se utilizarán así como sus ventajas de poder utilizar herramientas de software libre que permitan evolucionar a la plataforma y por ende a los proyectos relacionados con esta tecnología de desarrollo.

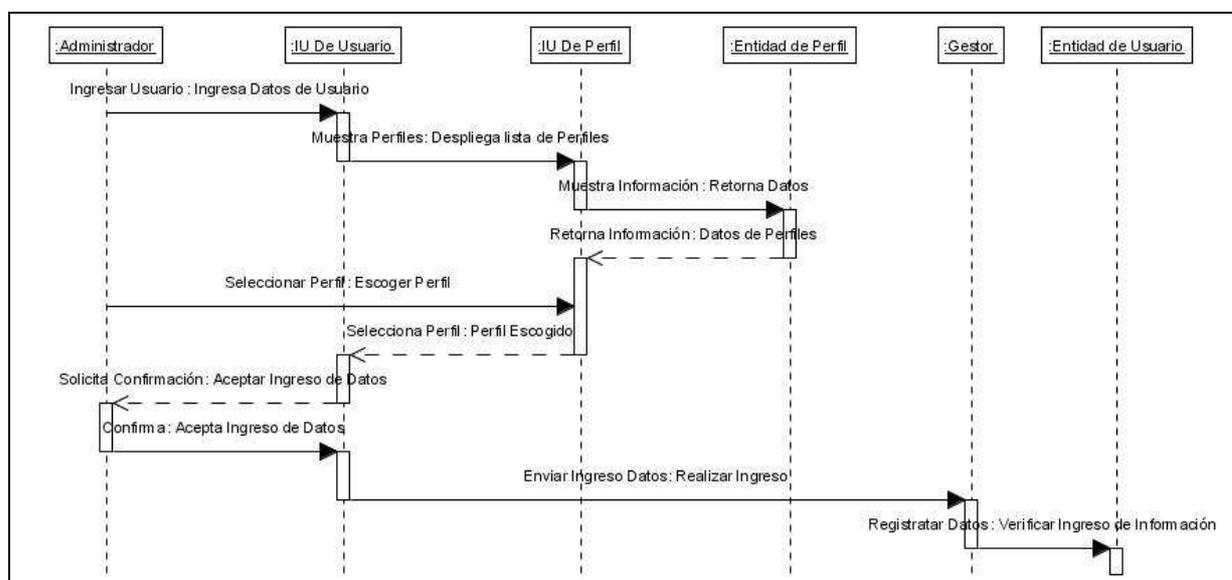
2.4.- DISEÑO

2.4.1.- MODELO DE DISEÑO

El modelo del diseño se basa en clases de diseño las cuales para su correcto análisis necesita como entrada al modelo de análisis, y en el cual se aprecio los requisitos funcionales de forma conceptual, teniendo esto como preámbulo modelaremos el sistema de colaboración a partir de la interacción entre objetos ya mas detallados con el uso de diagramas de secuencia los cuales mostrarán las acciones e interacciones mediante la transferencia de mensajes que existe en entre objetos.

2.4.2.- DIAGRAMAS DE SECUENCIA

MODELO:	Modelo de Diseño
DIAGRAMA:	Diagrama de Secuencia
CASO DE USO:	Gestión de Usuarios – Ingreso de Usuario
FIGURA:	Figura 2.41

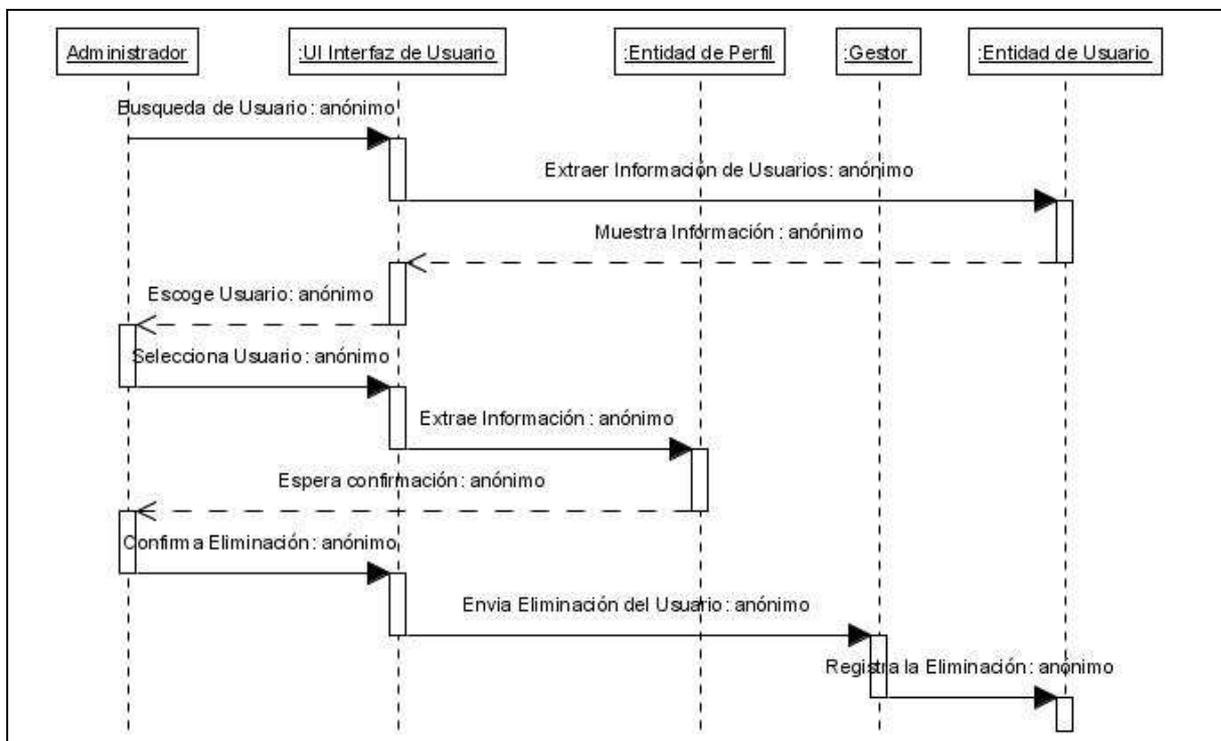


[Figura 2.41]

Elaborado por: El Autor

Referencia: Modelos desarrollados por el autor con la herramienta Poseidón.

MODELO:	Modelo de Diseño
DIAGRAMA:	Diagrama de Secuencia
CASO DE USO:	Gestión de Usuarios – Eliminación de Usuario
FIGURA:	Figura 2.42

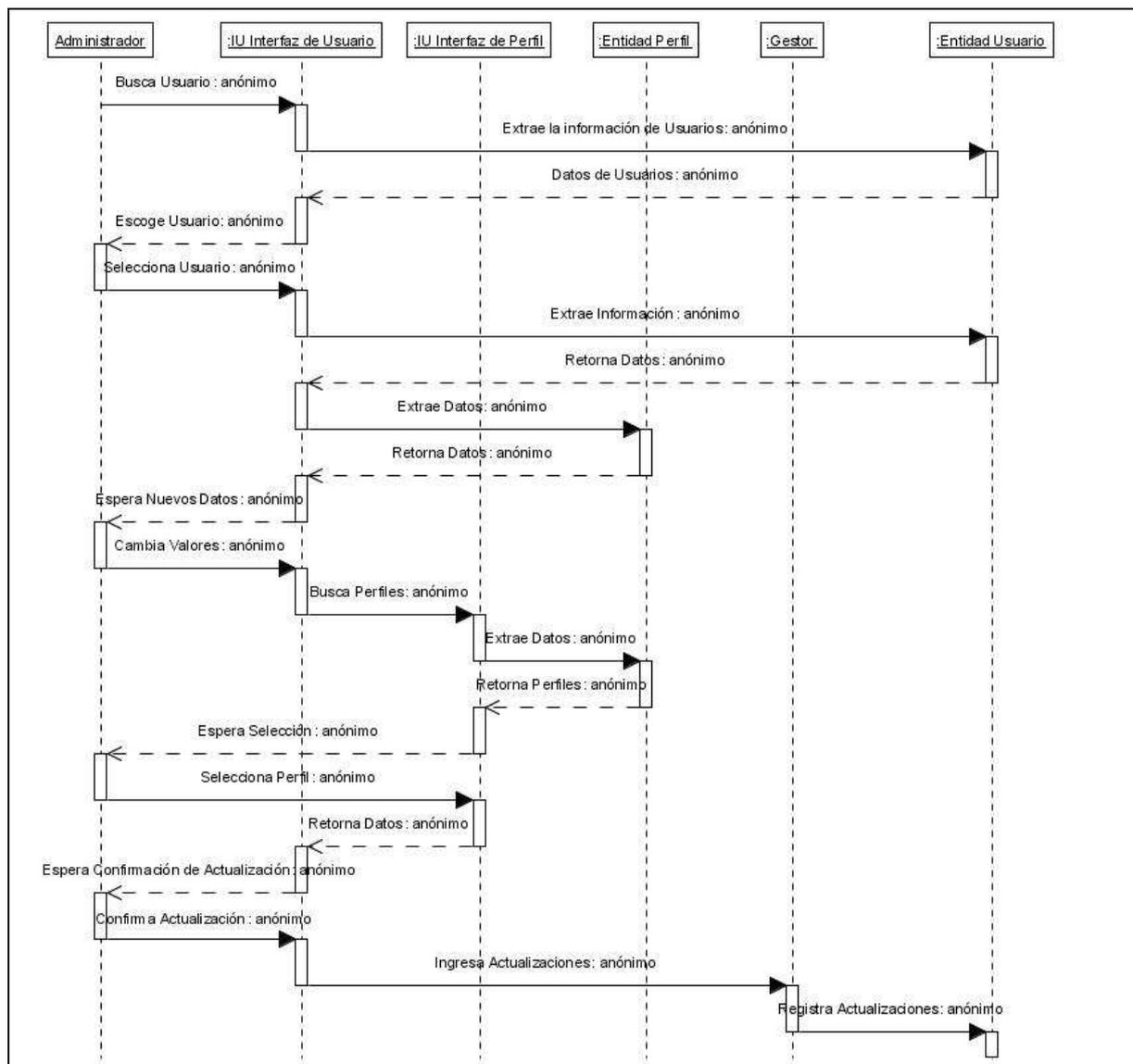


[Figura 2.42]

Elaborado por: El Autor

Referencia: Modelos desarrollados por el autor con la herramienta Poseidón.

MODELO:	Modelo de Diseño
DIAGRAMA:	Diagrama de Secuencia
CASO DE USO:	Gestión de Usuarios – Actualización de Usuario
FIGURA:	Figura 2.43

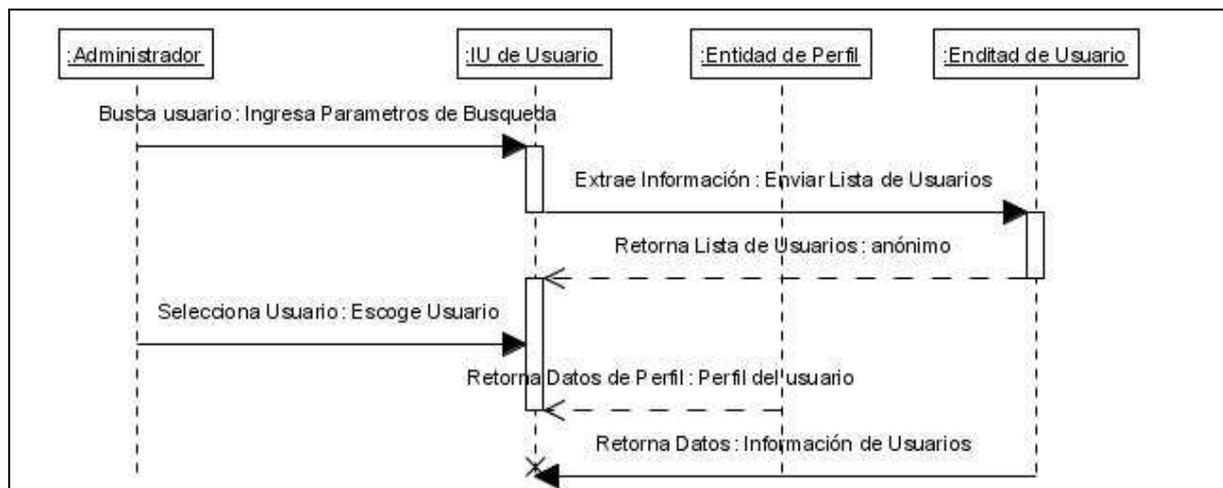


[Figura 2.43]

Elaborado por: El Autor

Referencia: Modelos desarrollados por el autor con la herramienta Poseidón.

MODELO:	Modelo de Diseño
DIAGRAMA:	Diagrama de Secuencia
CASO DE USO:	Gestión de Usuarios – Consulta de Usuario
FIGURA:	Figura 2.44

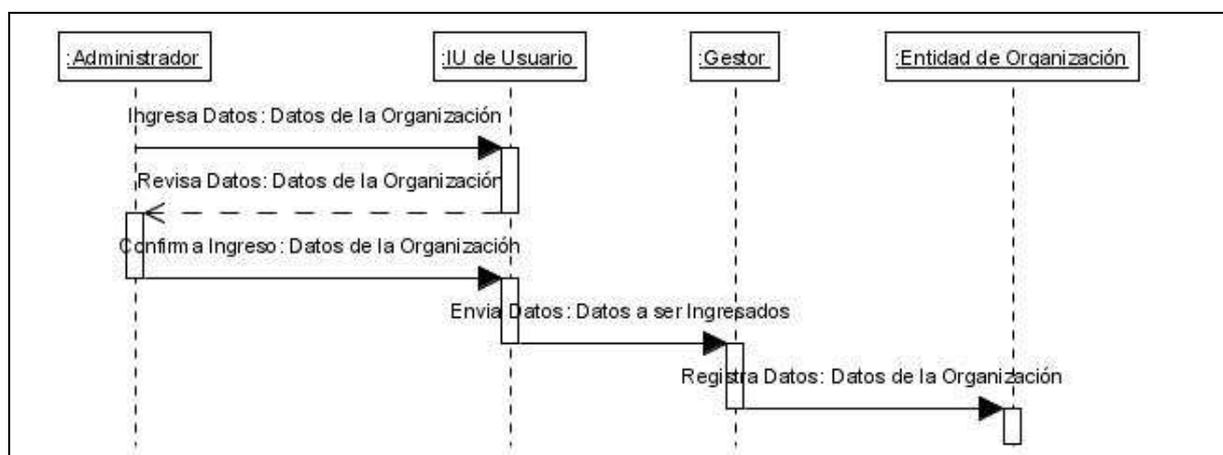


[Figura 2.44]

Elaborado por: El Autor

Referencia: Modelos desarrollados por el autor con la herramienta Poseidón.

MODELO:	Modelo de Diseño
DIAGRAMA:	Diagrama de Secuencia
CASO DE USO:	Gestión de la Organización – Ingreso de Datos Empresa
FIGURA:	Figura 2.45

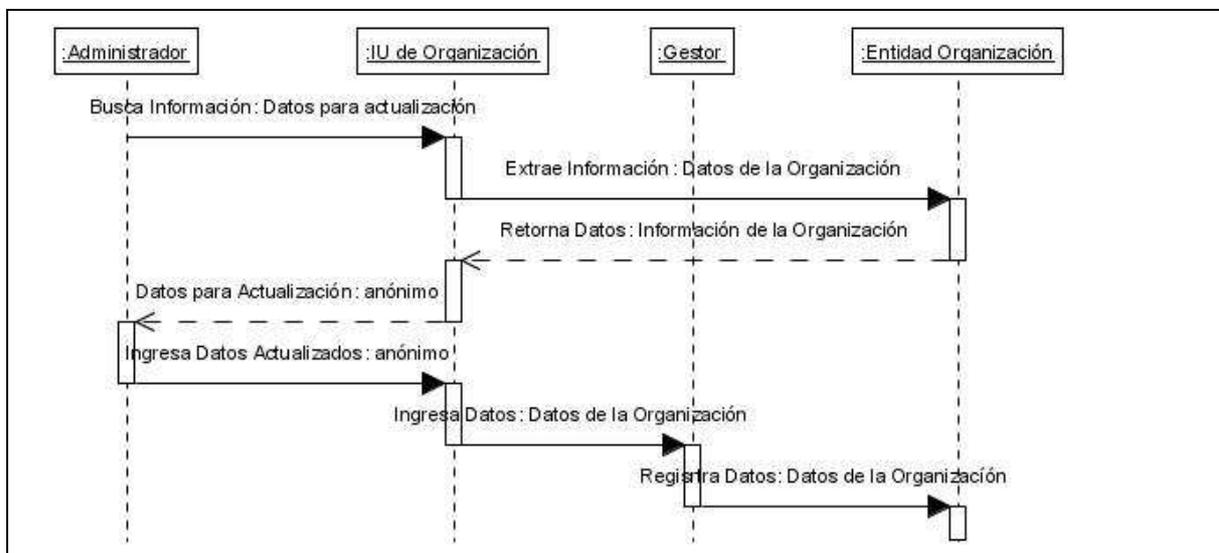


[Figura 2.45]

Elaborado por: El Autor

Referencia: Modelos desarrollados por el autor con la herramienta Poseidón.

MODELO:	Modelo de Diseño
DIAGRAMA:	Diagrama de Secuencia
CASO DE USO:	Gestión de la Organización – Actualización de Datos Empresa
FIGURA:	Figura 2.46

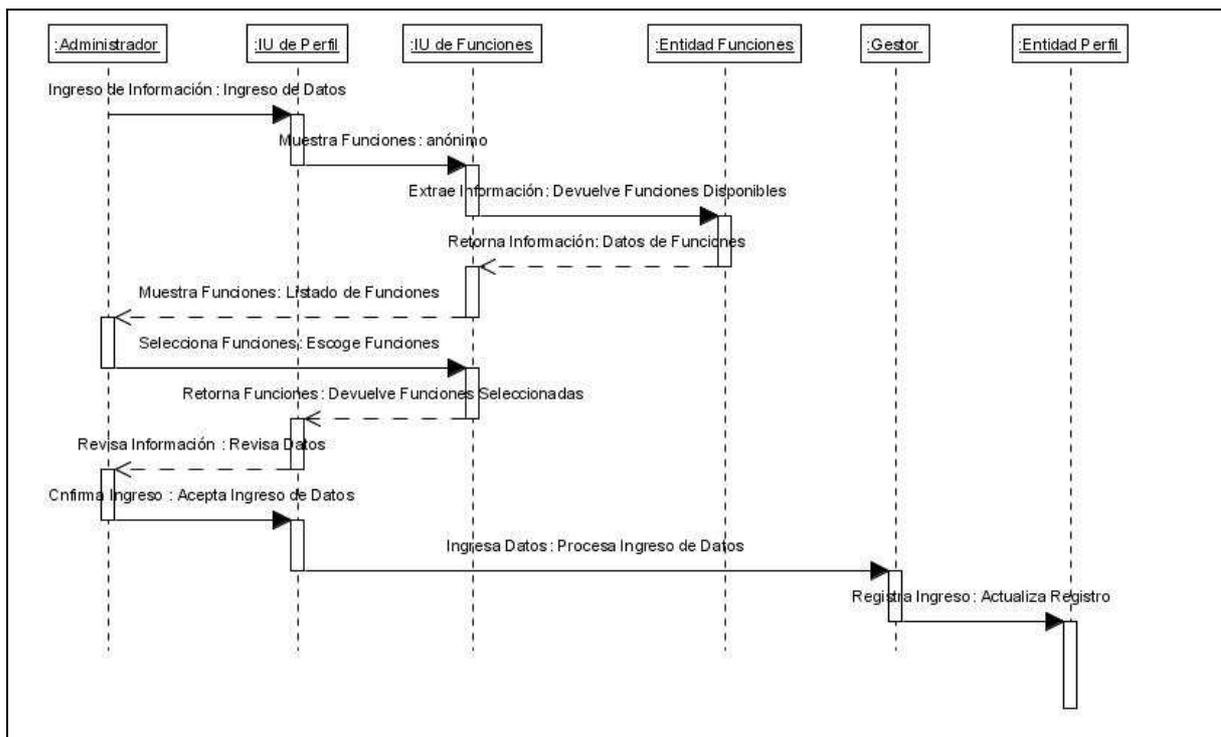


[Figura 2.46]

Elaborado por: El Autor

Referencia: Modelos desarrollados por el autor con la herramienta Poseidón.

MODELO:	Modelo de Diseño
DIAGRAMA:	Diagrama de Secuencia
CASO DE USO:	Gestión de Perfiles – Ingreso de Perfil
FIGURA:	Figura 2.47

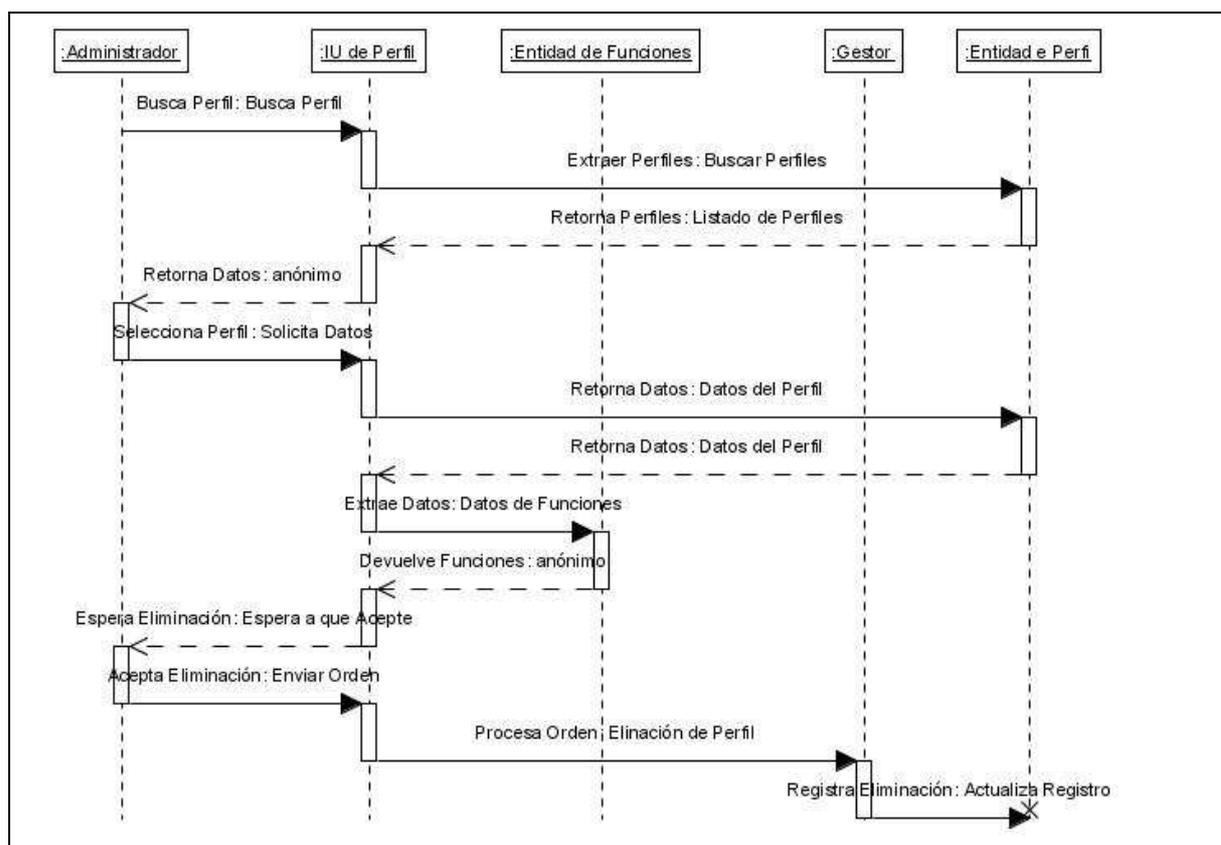


[Figura 2.47]

Elaborado por: El Autor

Referencia: Modelos desarrollados por el autor con la herramienta Poseidón.

MODELO:	Modelo de Diseño
DIAGRAMA:	Diagrama de Secuencia
CASO DE USO:	Gestión de Perfiles – Eliminación de Perfil
FIGURA:	Figura 2.48

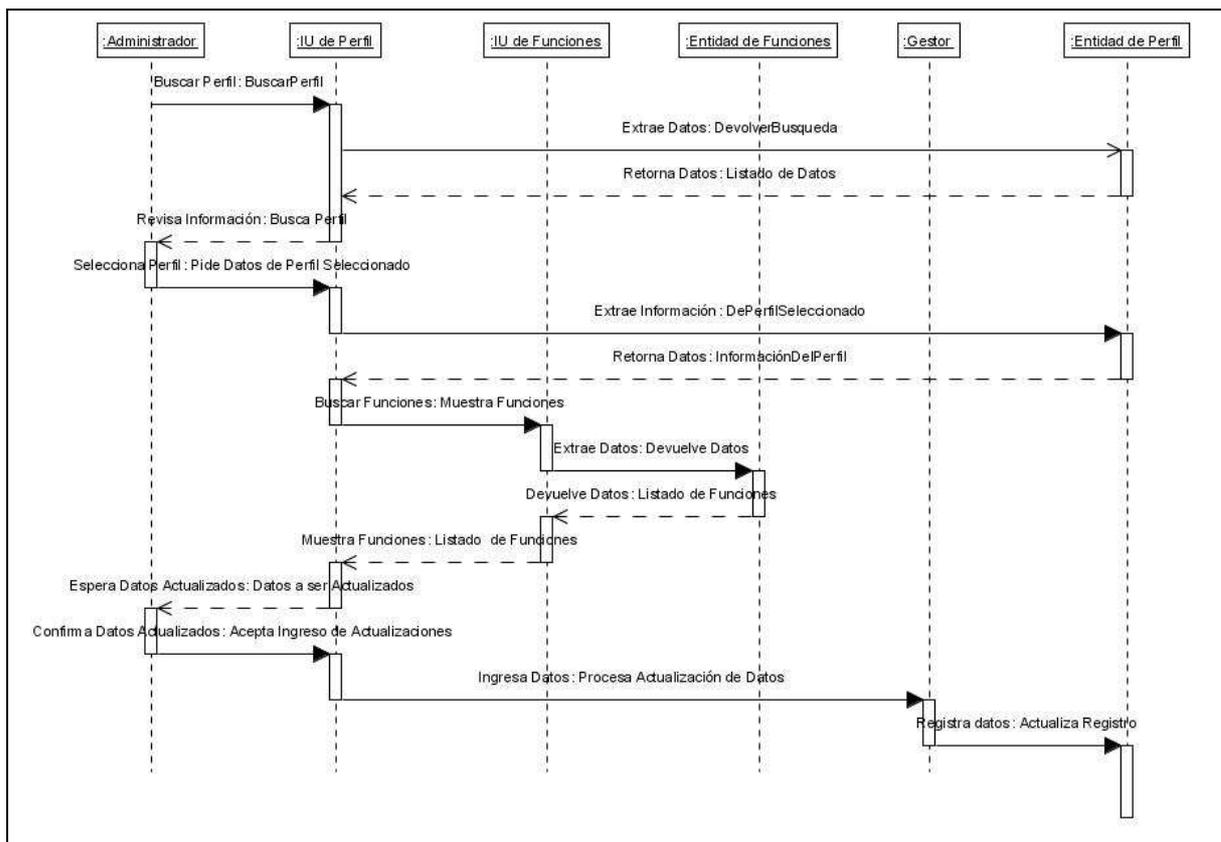


[Figura 2.48]

Elaborado por: El Autor

Referencia: Modelos desarrollados por el autor con la herramienta Poseidón.

MODELO:	Modelo de Diseño
DIAGRAMA:	Diagrama de Secuencia
CASO DE USO:	Gestión de Perfiles – Actualización de Perfil
FIGURA:	Figura 2.49

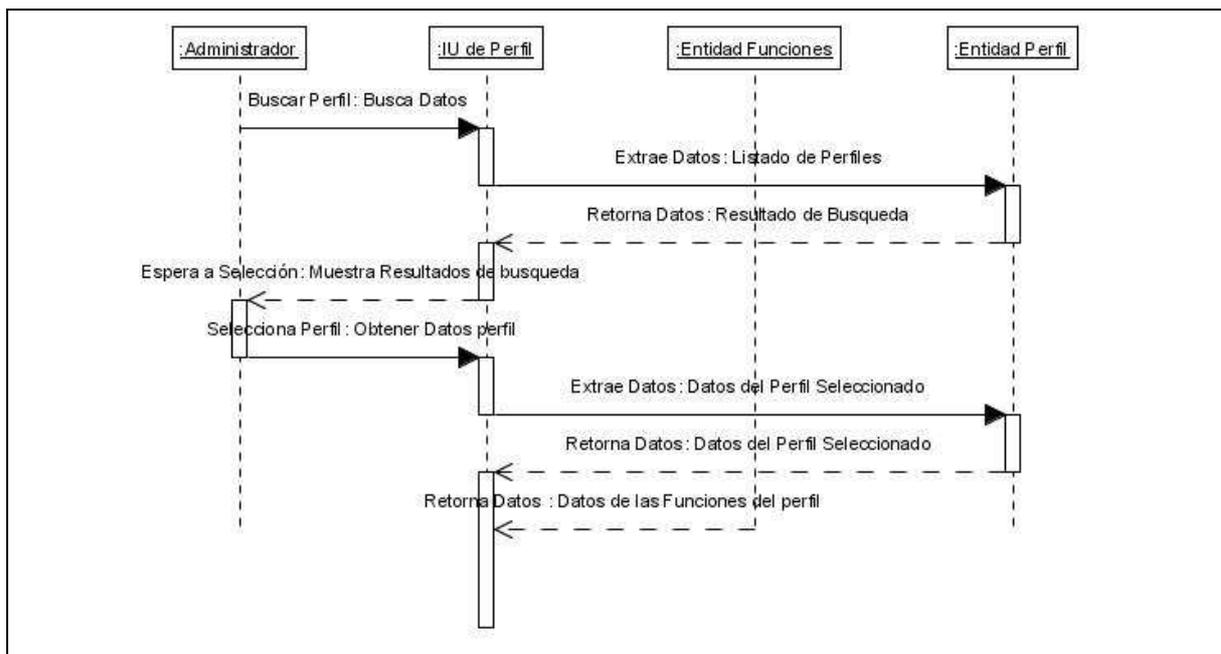


[Figura 2.49]

Elaborado por: El Autor

Referencia: Modelos desarrollados por el autor con la herramienta Poseidón.

MODELO:	Modelo de Diseño
DIAGRAMA:	Diagrama de Secuencia
CASO DE USO:	Gestión de Perfiles – Consulta de Perfil
FIGURA:	Figura 2.50

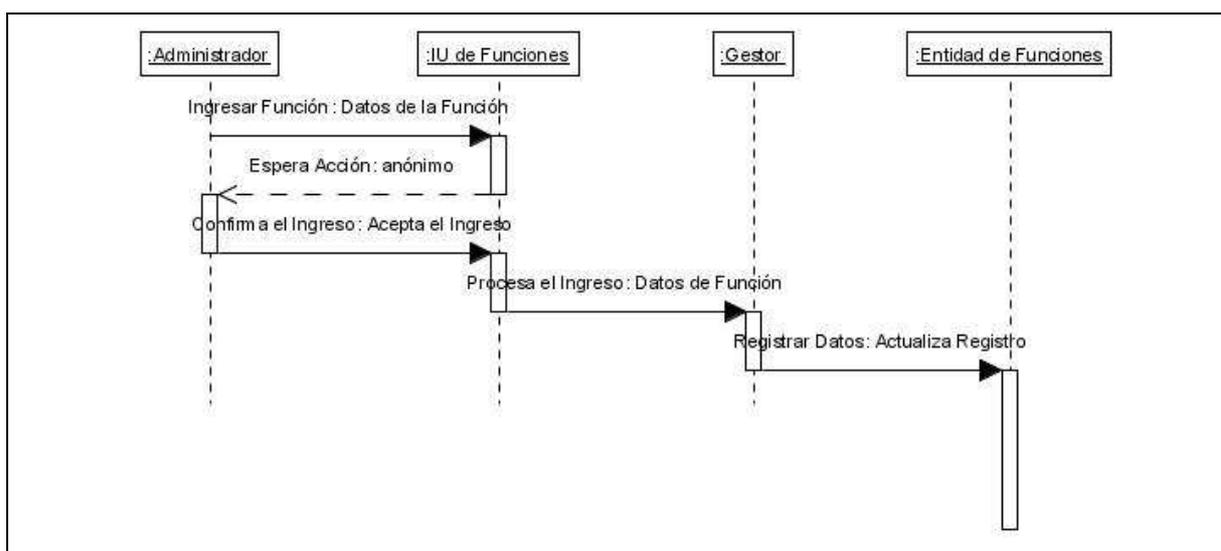


[Figura 2.50]

Elaborado por: El Autor

Referencia: Modelos desarrollados por el autor con la herramienta Poseidón.

MODELO:	Modelo de Diseño
DIAGRAMA:	Diagrama de Secuencia
CASO DE USO:	Gestión de Funciones – Ingreso de Función
FIGURA:	Figura 2.51

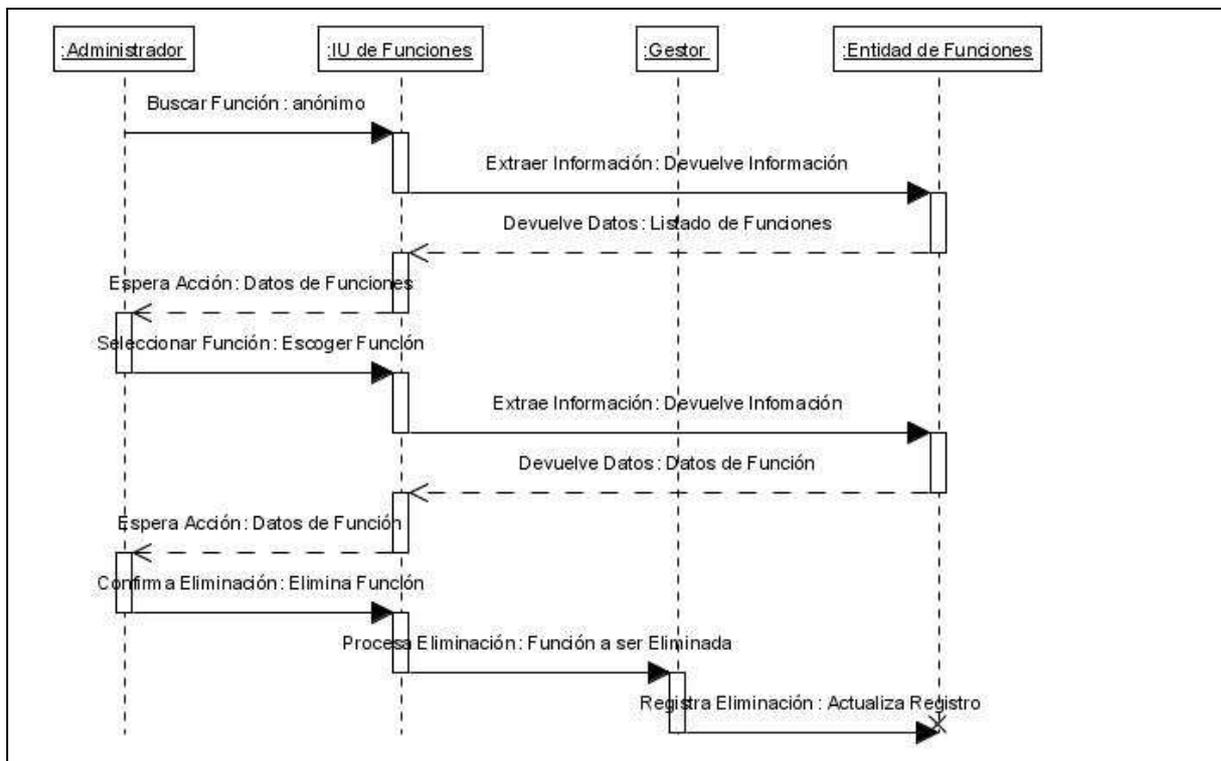


[Figura 2.51]

Elaborado por: El Autor

Referencia: Modelos desarrollados por el autor con la herramienta Poseidón.

MODELO:	Modelo de Diseño
DIAGRAMA:	Diagrama de Secuencia
CASO DE USO:	Gestión de Funciones – Eliminación de Función
FIGURA:	Figura 2.52

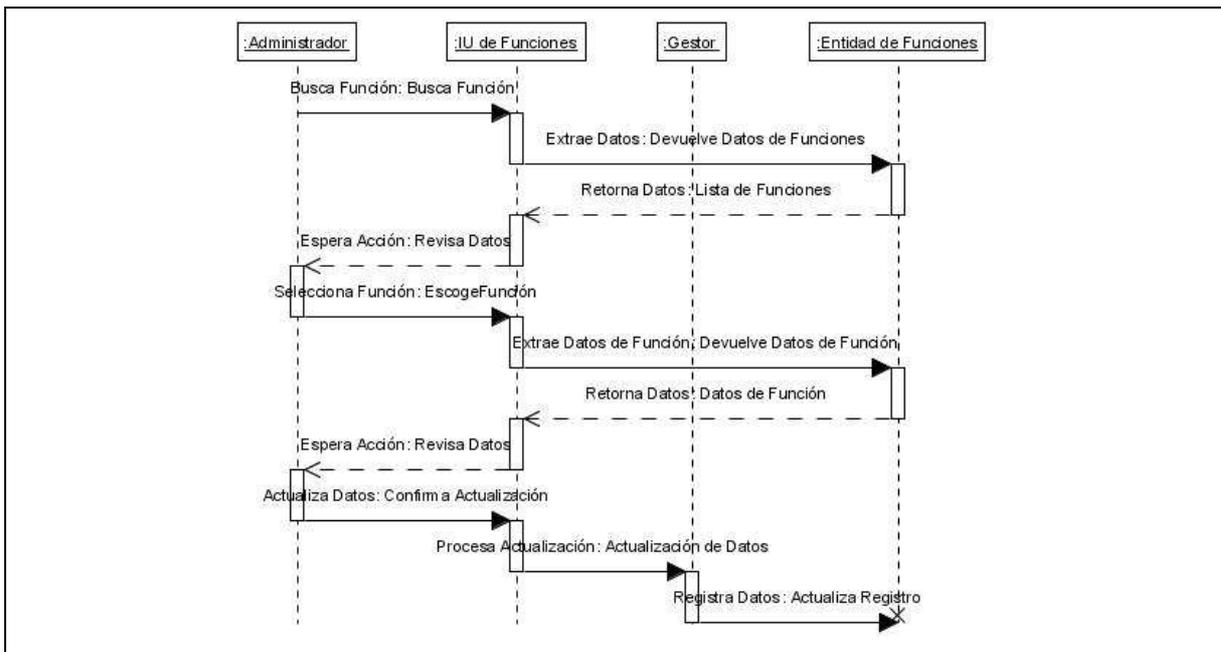


[Figura 2.52]

Elaborado por: El Autor

Referencia: Modelos desarrollados por el autor con la herramienta Poseidón.

MODELO:	Modelo de Diseño
DIAGRAMA:	Diagrama de Secuencia
CASO DE USO:	Gestión de Funciones – Actualización de Función
FIGURA:	Figura 2.53

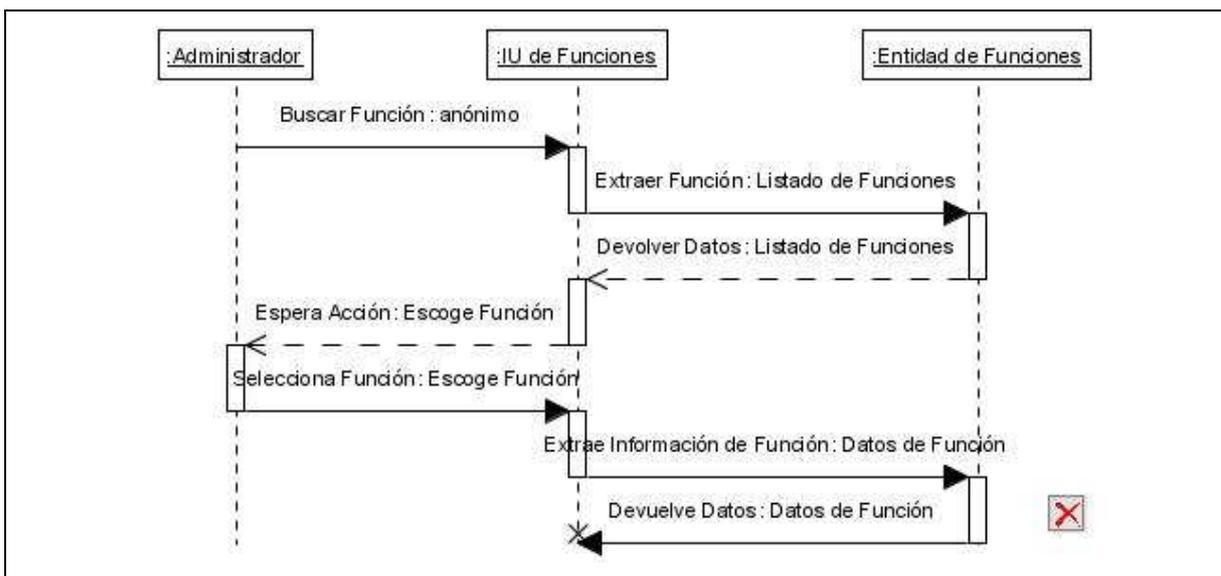


[Figura 2.53]

Elaborado por: El Autor

Referencia: Modelos desarrollados por el autor con la herramienta Poseidón.

MODELO:	Modelo de Diseño
DIAGRAMA:	Diagrama de Secuencia
CASO DE USO:	Gestión de Funciones – Consulta de Función
FIGURA:	Figura 2.54

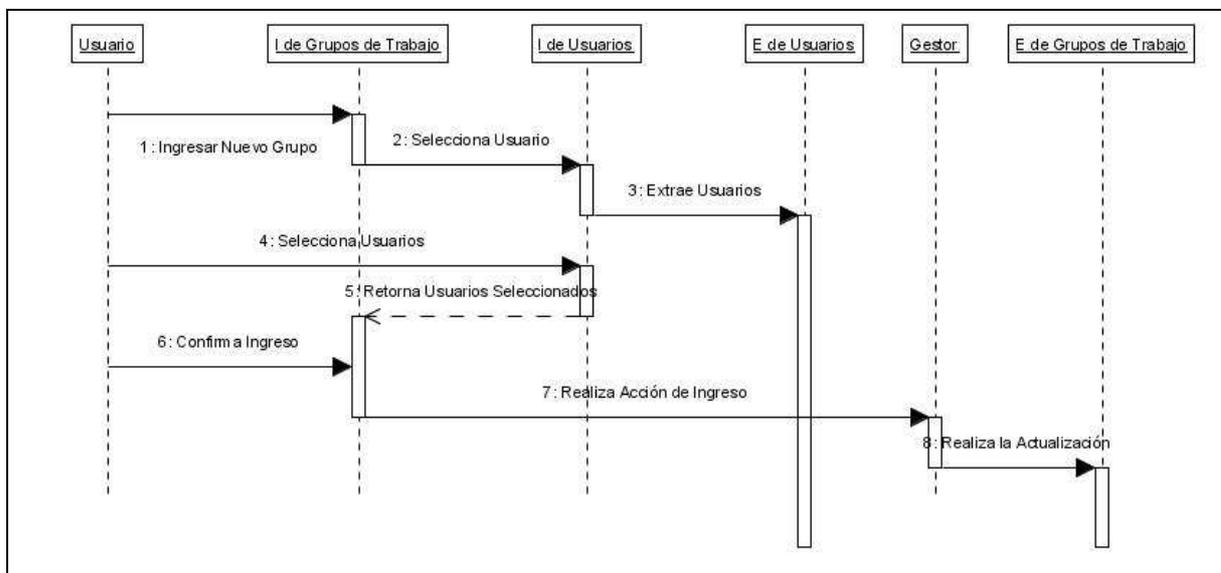


[Figura 2.54]

Elaborado por: El Autor

Referencia: Modelos desarrollados por el autor con la herramienta Poseidón.

MODELO:	Modelo de Diseño
DIAGRAMA:	Diagrama de Secuencia
CASO DE USO:	Gestión de Grupos de Trabajo – Ingreso de Grupo
FIGURA:	Figura 2.55

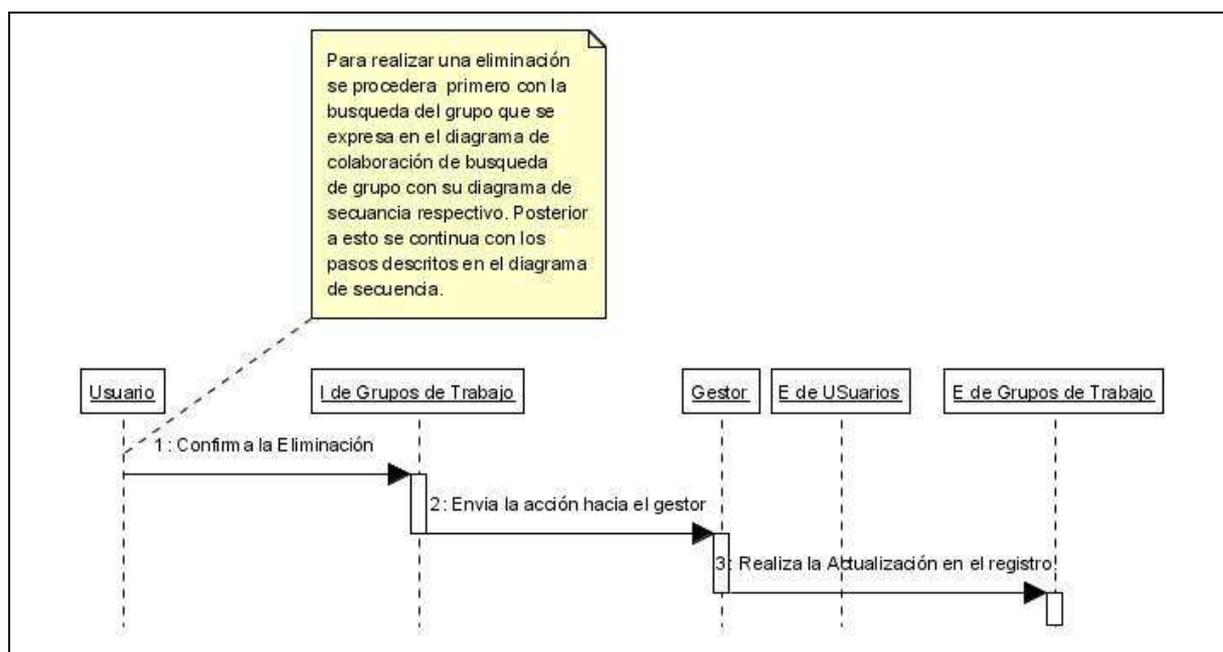


[Figura 2.55]

Elaborado por: El Autor

Referencia: Modelos desarrollados por el autor con la herramienta Poseidón.

MODELO:	Modelo de Diseño
DIAGRAMA:	Diagrama de Secuencia
CASO DE USO:	Gestión de Grupos de Trabajo – Eliminación de Grupo
FIGURA:	Figura 2.56

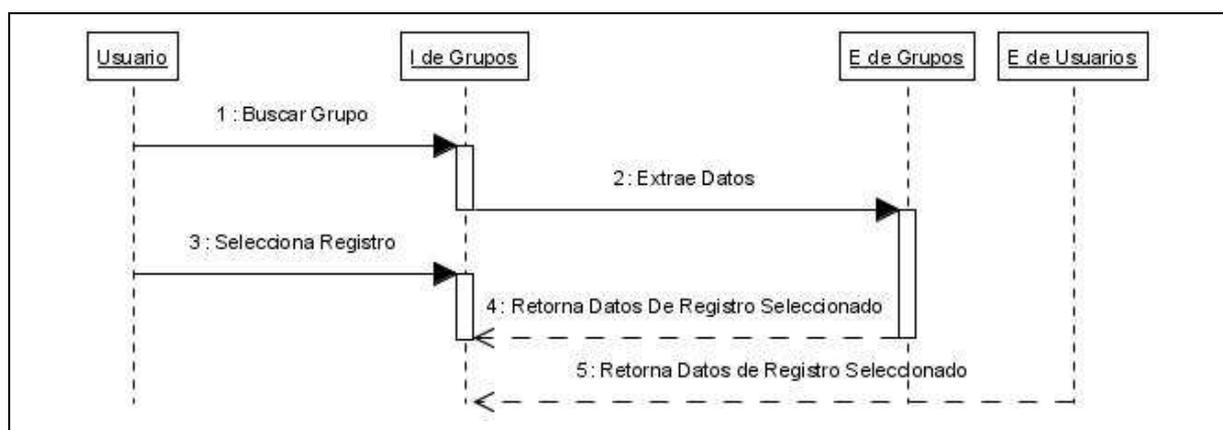


[Figura 2.56]

Elaborado por: El Autor

Referencia: Modelos desarrollados por el autor con la herramienta Poseidón.

MODELO:	Modelo de Diseño
DIAGRAMA:	Diagrama de Secuencia
CASO DE USO:	Gestión de Grupos de Trabajo – Consulta de Grupos
FIGURA:	Figura 2.57

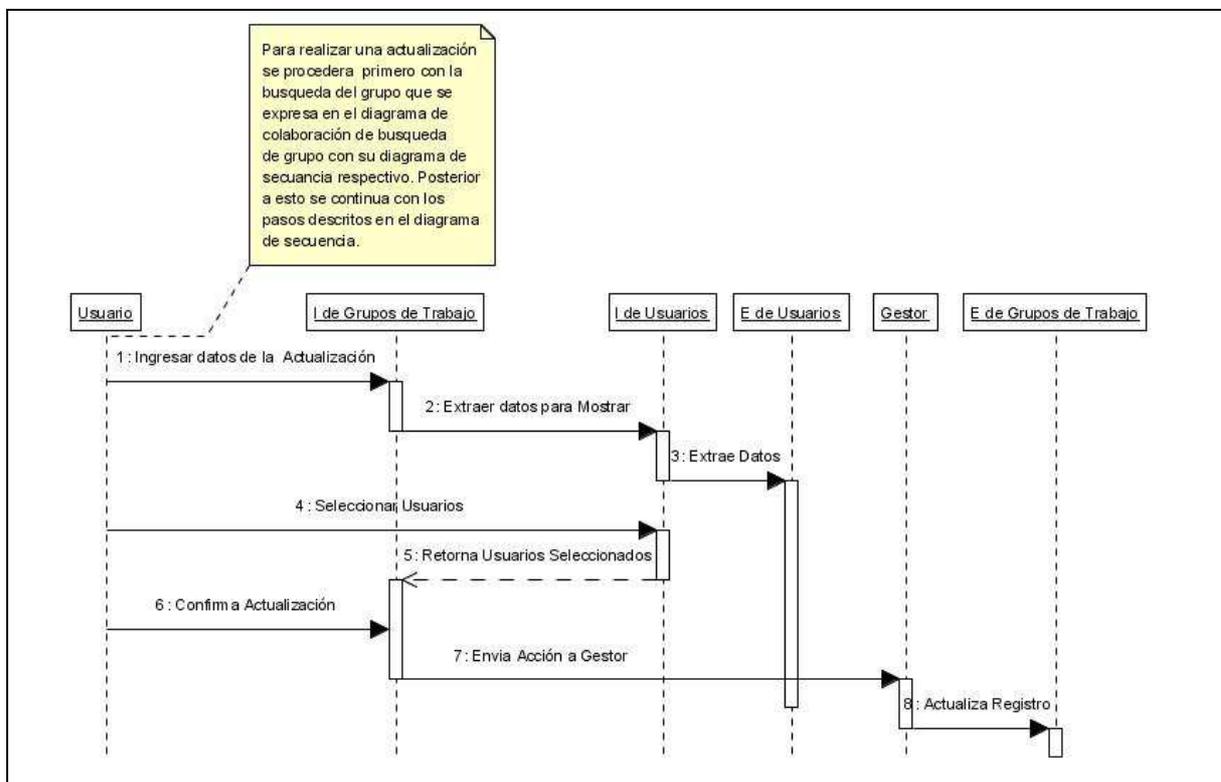


[Figura 2.57]

Elaborado por: El Autor

Referencia: Modelos desarrollados por el autor con la herramienta Poseidón.

MODELO:	Modelo de Diseño
DIAGRAMA:	Diagrama de Secuencia
CASO DE USO:	Gestión de Grupos de Trabajo – Actualización de Grupo
FIGURA:	Figura 2.58

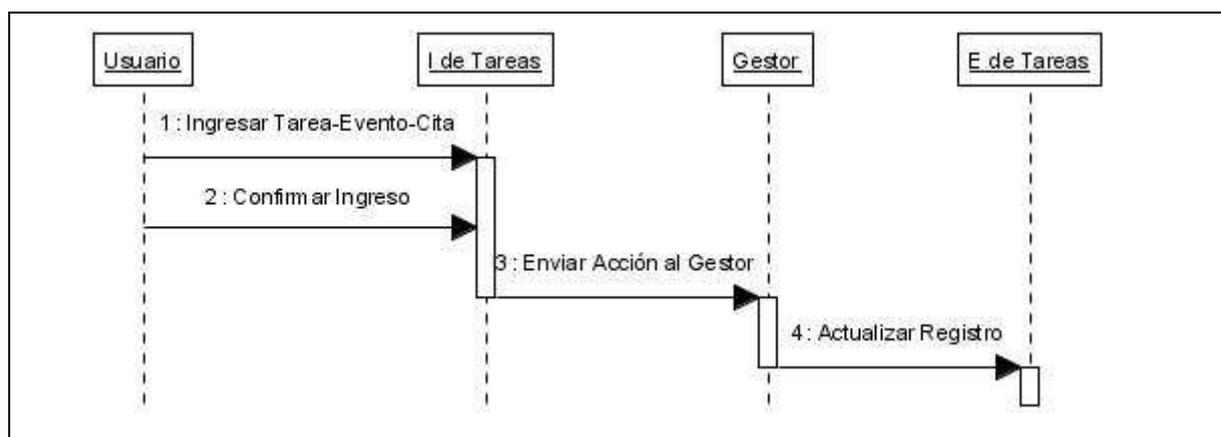


[Figura 2.58]

Elaborado por: El Autor

Referencia: Modelos desarrollados por el autor con la herramienta Poseidón.

MODELO:	Modelo de Diseño
DIAGRAMA:	Diagrama de Secuencia
CASO DE USO:	Gestión de Tareas – Ingreso de Tareas
FIGURA:	Figura 2.59

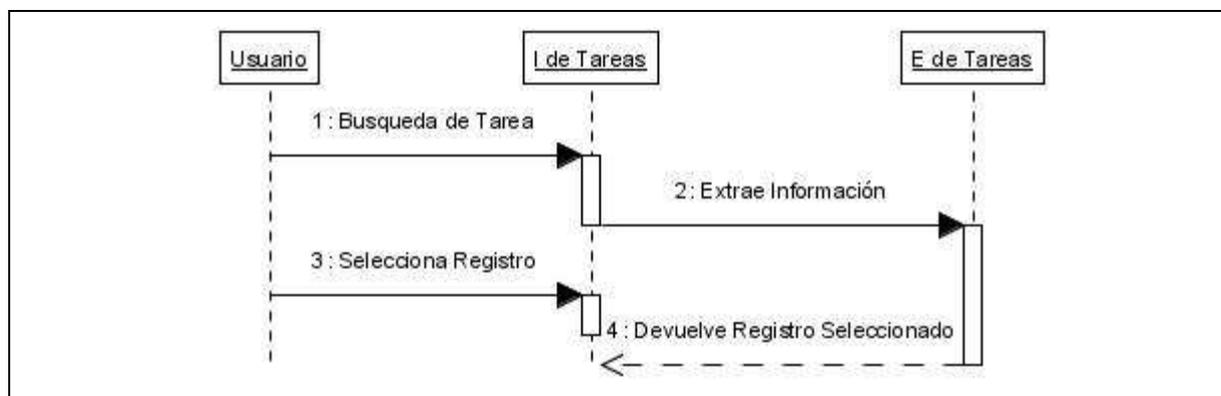


[Figura 2.59]

Elaborado por: El Autor

Referencia: Modelos desarrollados por el autor con la herramienta Poseidón.

MODELO:	Modelo de Diseño
DIAGRAMA:	Diagrama de Secuencia
CASO DE USO:	Gestión de Tareas – Consulta de Tareas
FIGURA:	Figura 2.60

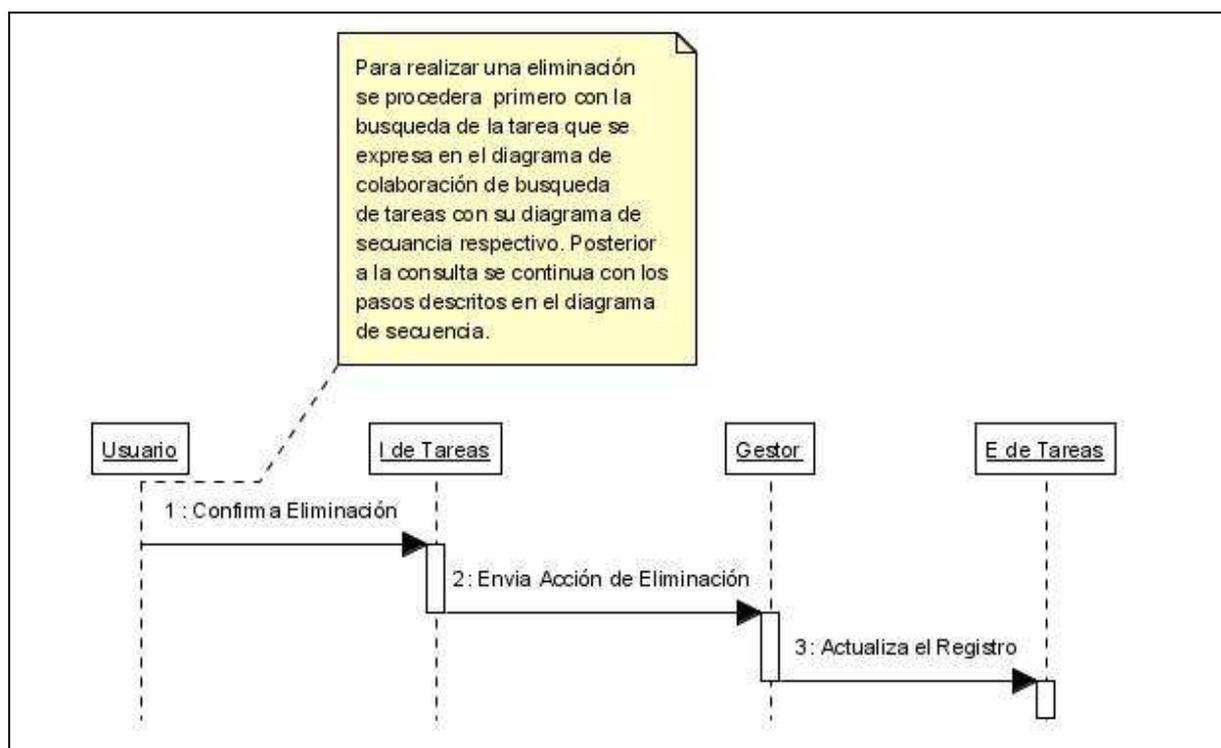


[Figura 2.60]

Elaborado por: El Autor

Referencia: Modelos desarrollados por el autor con la herramienta Poseidón.

MODELO:	Modelo de Diseño
DIAGRAMA:	Diagrama de Secuencia
CASO DE USO:	Gestión de Tareas – Actualización de Tareas
FIGURA:	Figura 2.61

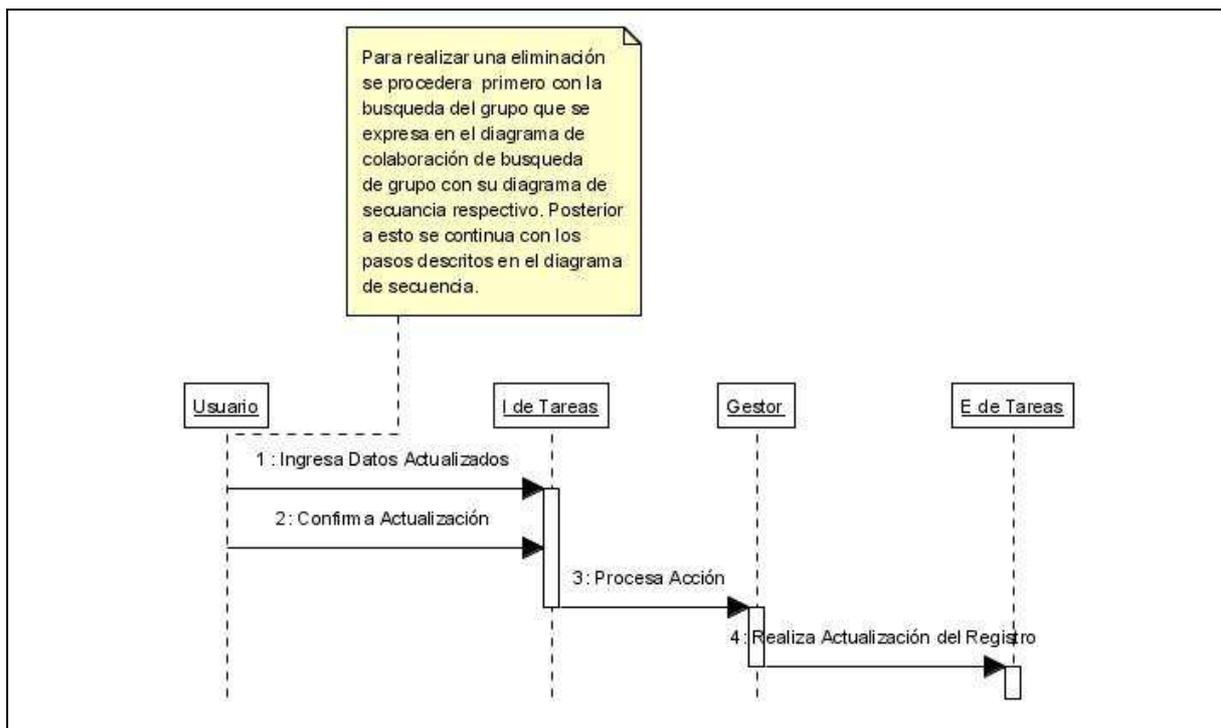


[Figura 2.61]

Elaborado por: El Autor

Referencia: Modelos desarrollados por el autor con la herramienta Poseidón.

MODELO:	Modelo de Diseño
DIAGRAMA:	Diagrama de Secuencia
CASO DE USO:	Gestión de Tareas – Eliminación de Tareas
FIGURA:	Figura 2.62

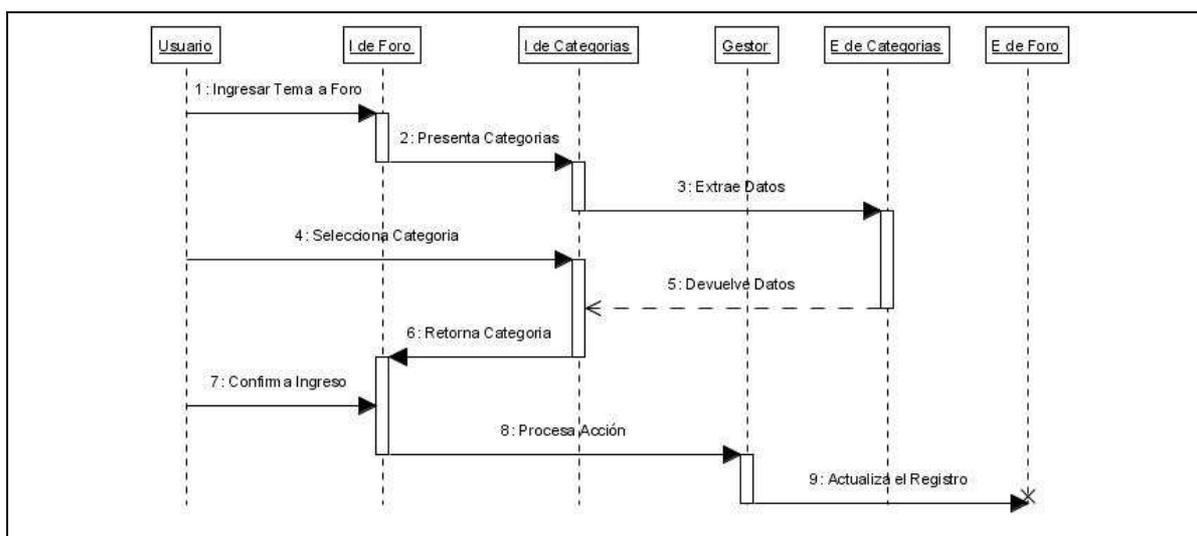


[Figura 2.62]

Elaborado por: El Autor

Referencia: Modelos desarrollados por el autor con la herramienta Poseidón.

MODELO:	Modelo de Diseño
DIAGRAMA:	Diagrama de Secuencia
CASO DE USO:	Gestión de Foros de Discusión – Ingresar Tema
FIGURA:	Figura 2.63

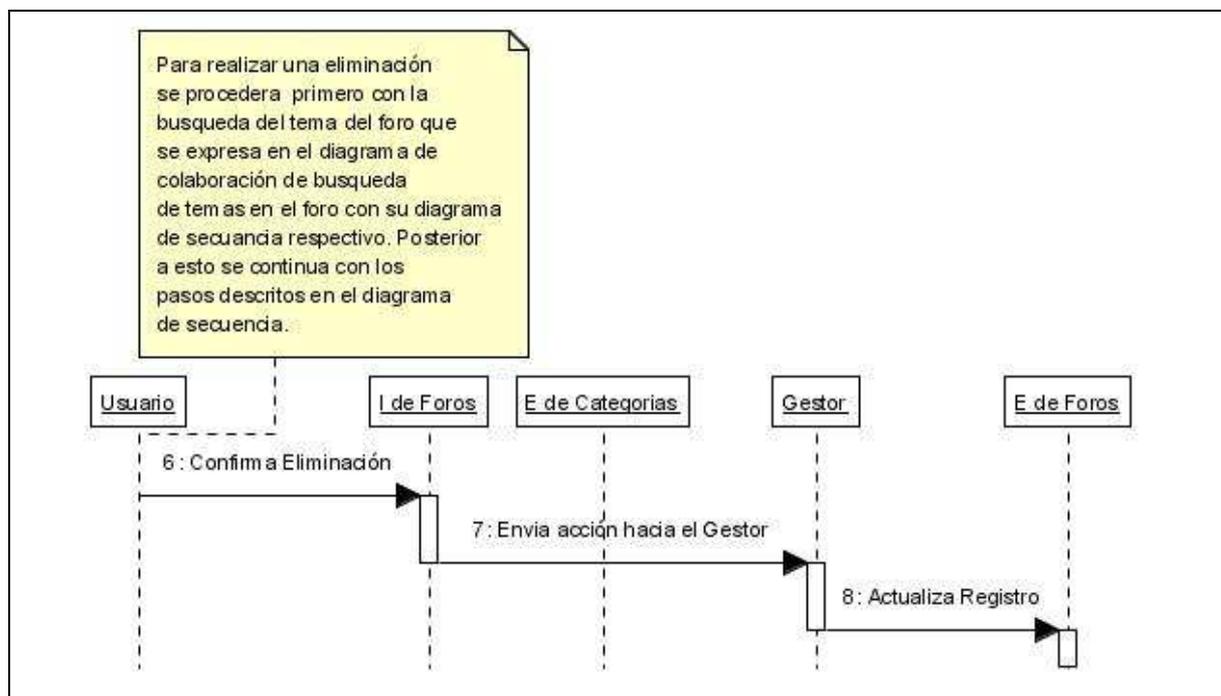


[Figura 2.63]

Elaborado por: El Autor

Referencia: Modelos desarrollados por el autor con la herramienta Poseidón.

MODELO:	Modelo de Diseño
DIAGRAMA:	Diagrama de Secuencia
CASO DE USO:	Gestión de Foros de Discusión – Eliminar Tema
FIGURA:	Figura 2.64

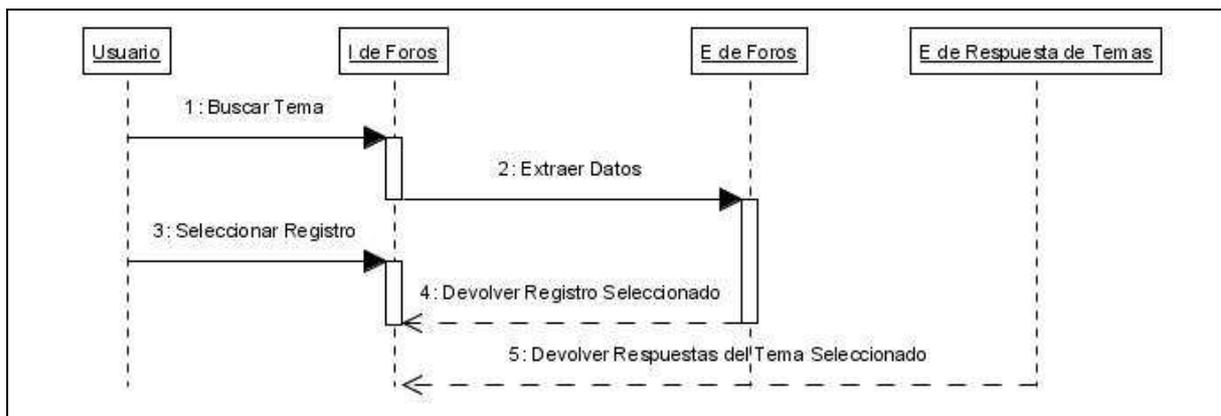


[Figura 2.64]

Elaborado por: El Autor

Referencia: Modelos desarrollados por el autor con la herramienta Poseidón.

MODELO:	Modelo de Diseño
DIAGRAMA:	Diagrama de Secuencia
CASO DE USO:	Gestión de Foros de Discusión – Consultar Tema
FIGURA:	Figura 2.65

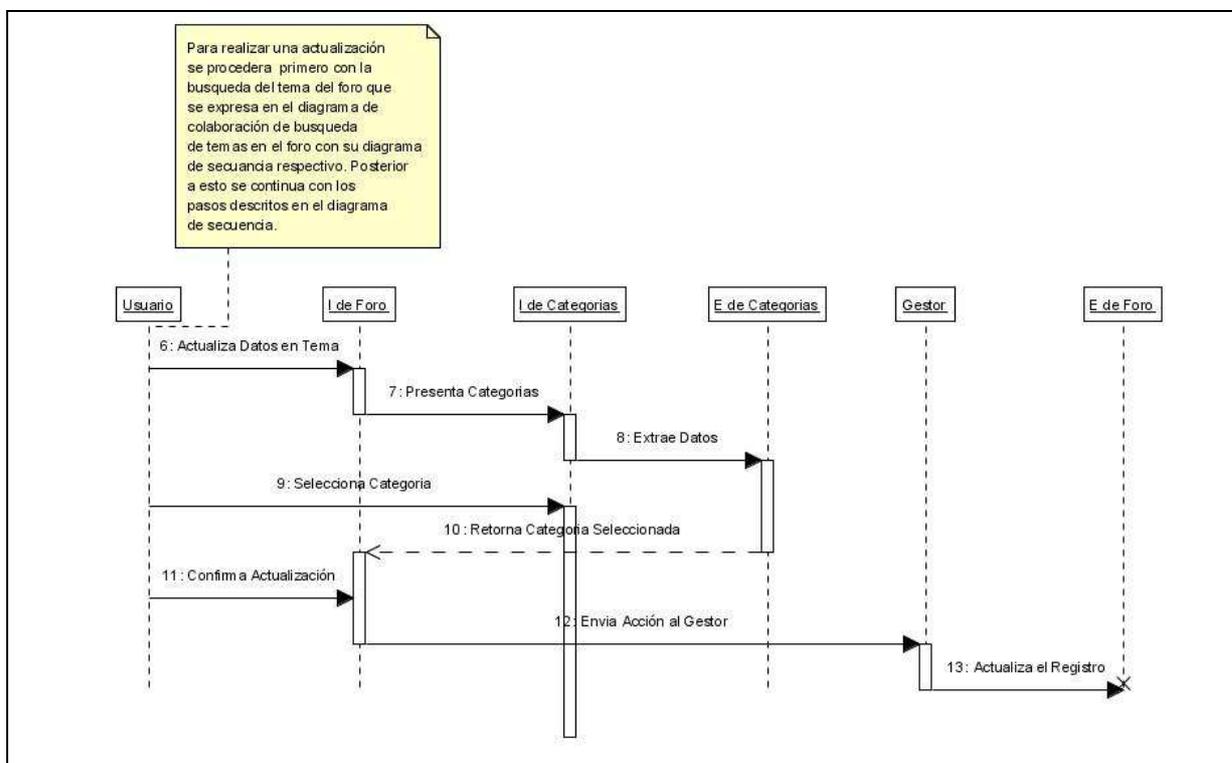


[Figura 2.65]

Elaborado por: El Autor

Referencia: Modelos desarrollados por el autor con la herramienta Poseidón.

MODELO:	Modelo de Diseño
DIAGRAMA:	Diagrama de Secuencia
CASO DE USO:	Gestión de Foros de Discusión – Actualizar Tema
FIGURA:	Figura 2.66

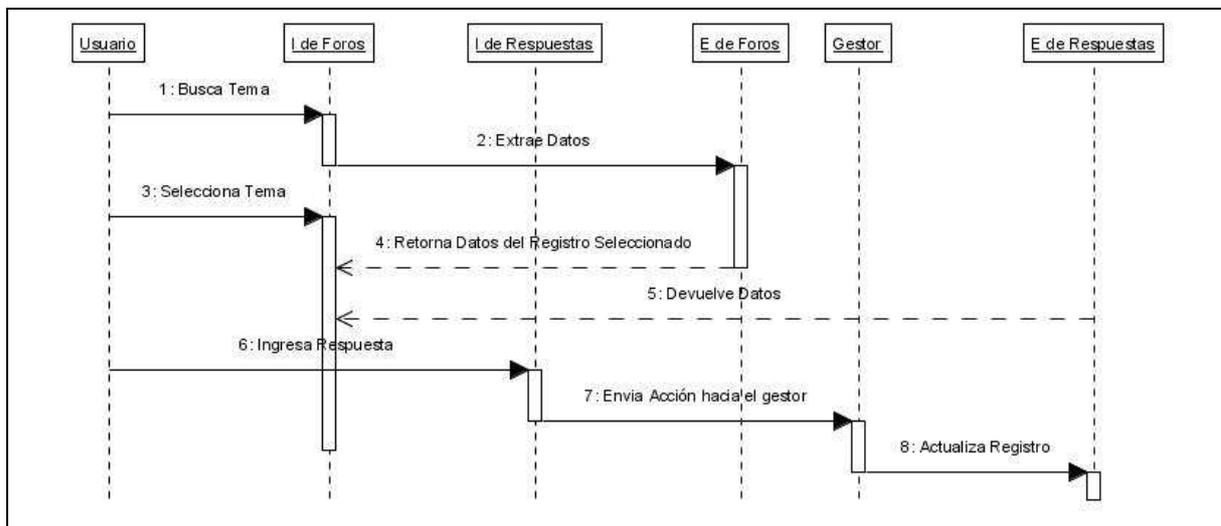


[Figura 2.66]

Elaborado por: El Autor

Referencia: Modelos desarrollados por el autor con la herramienta Poseidón.

MODELO:	Modelo de Diseño
DIAGRAMA:	Diagrama de Secuencia
CASO DE USO:	Gestión de Foros de Discusión – Ingresar respuesta a Tema
FIGURA:	Figura 2.67

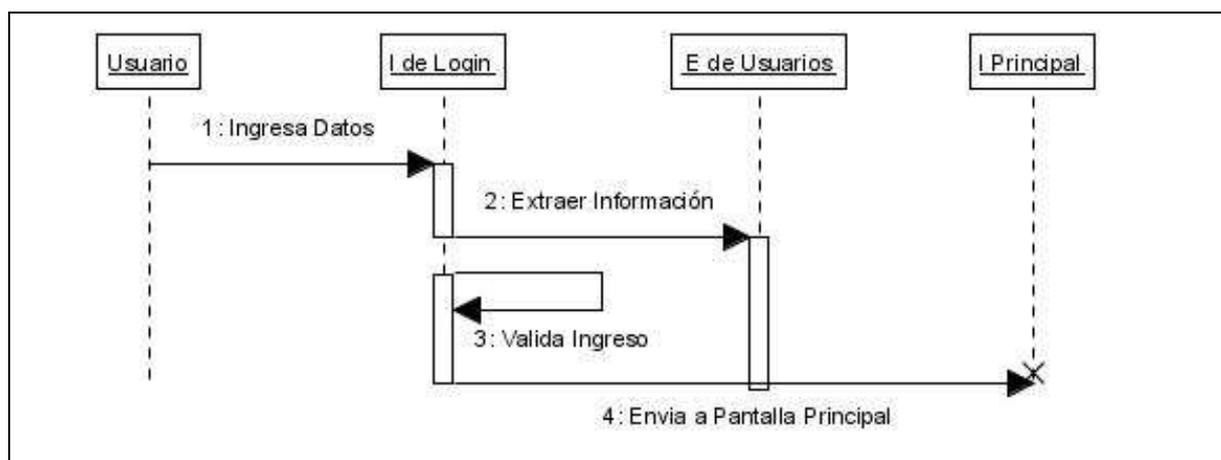


[Figura 2.67]

Elaborado por: El Autor

Referencia: Modelos desarrollados por el autor con la herramienta Poseidón.

MODELO:	Modelo de Diseño
DIAGRAMA:	Diagrama de Secuencia
CASO DE USO:	Ingreso de Usuario al sistema.
FIGURA:	Figura 2.68

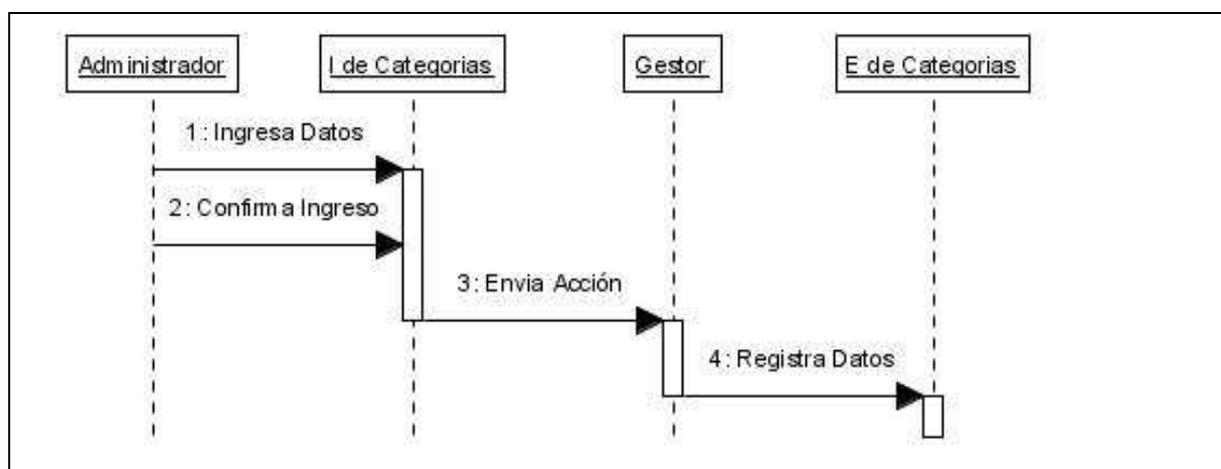


[Figura 2.68]

Elaborado por: El Autor

Referencia: Modelos desarrollados por el autor con la herramienta Poseidón.

MODELO:	Modelo de Diseño
DIAGRAMA:	Diagrama de Secuencia
CASO DE USO:	Gestión de Categorías de Foros – Ingreso de Categoría
FIGURA:	Figura 2.69

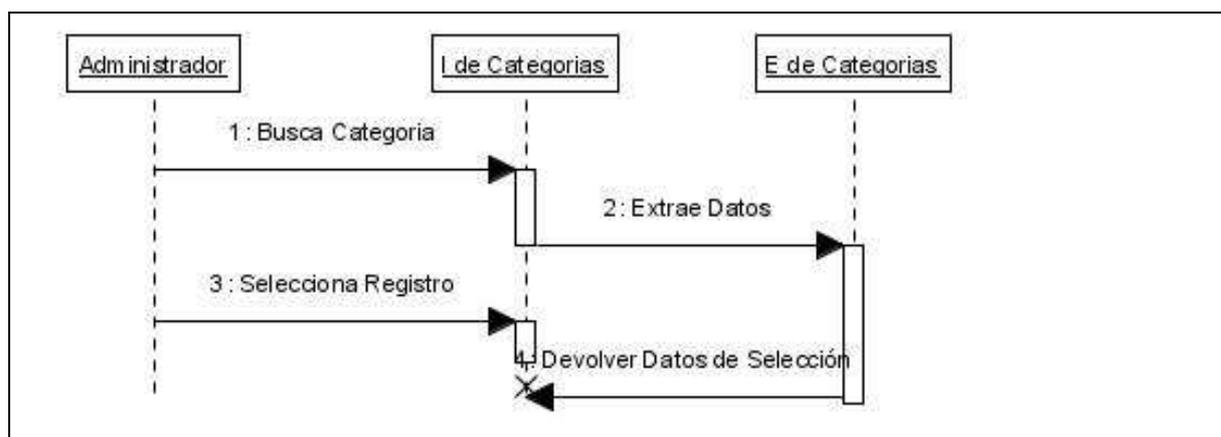


[Figura 2.69]

Elaborado por: El Autor

Referencia: Modelos desarrollados por el autor con la herramienta Poseidón.

MODELO:	Modelo de Diseño
DIAGRAMA:	Diagrama de Secuencia
CASO DE USO:	Gestión de Categorías de Foros – Consultar Categoría
FIGURA:	Figura 2.70

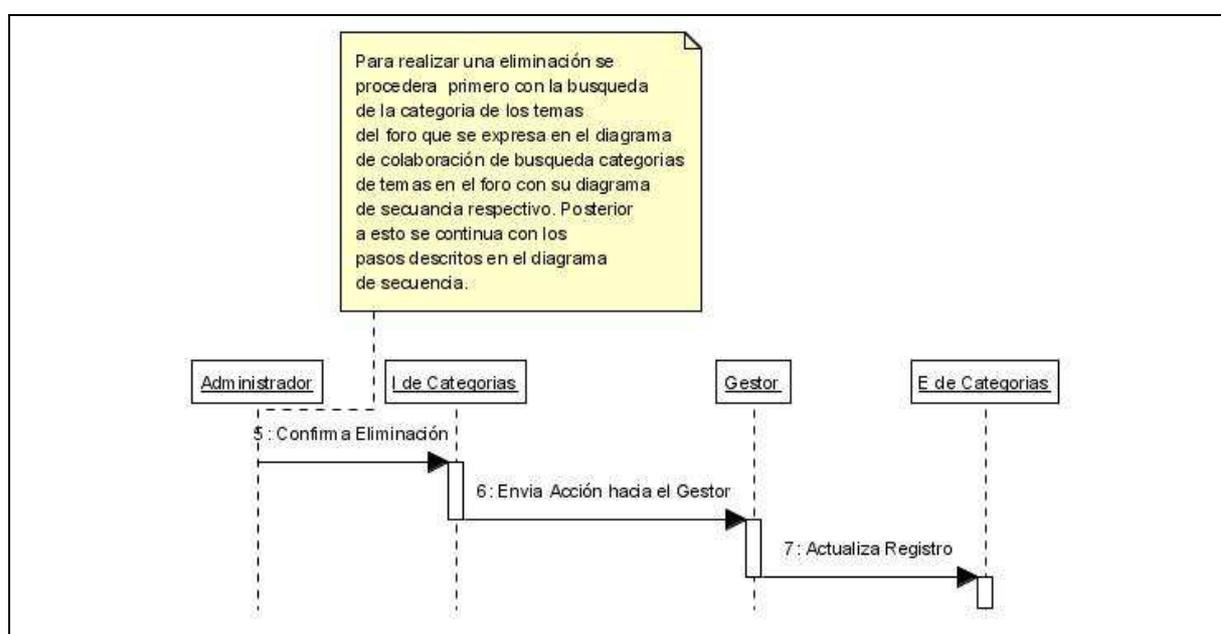


[Figura 2.70]

Elaborado por: El Autor

Referencia: Modelos desarrollados por el autor con la herramienta Poseidón.

MODELO:	Modelo de Diseño
DIAGRAMA:	Diagrama de Secuencia
CASO DE USO:	Gestión de Categorías de Foros – Eliminación de Categoría
FIGURA:	Figura 2.71

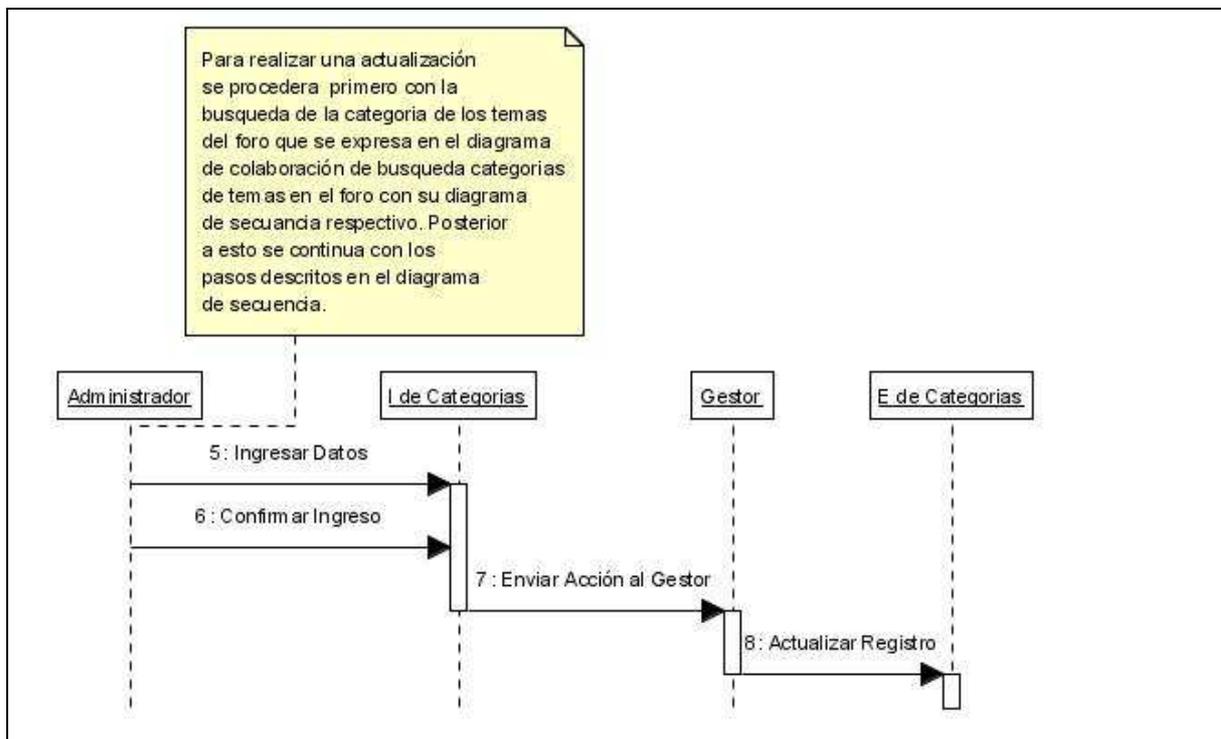


[Figura 2.71]

Elaborado por: El Autor

Referencia: Modelos desarrollados por el autor con la herramienta Poseidón.

MODELO:	Modelo de Diseño
DIAGRAMA:	Diagrama de Secuencia
CASO DE USO:	Gestión de Categorías de Foros – Actualización de Categoría
FIGURA:	Figura 2.72



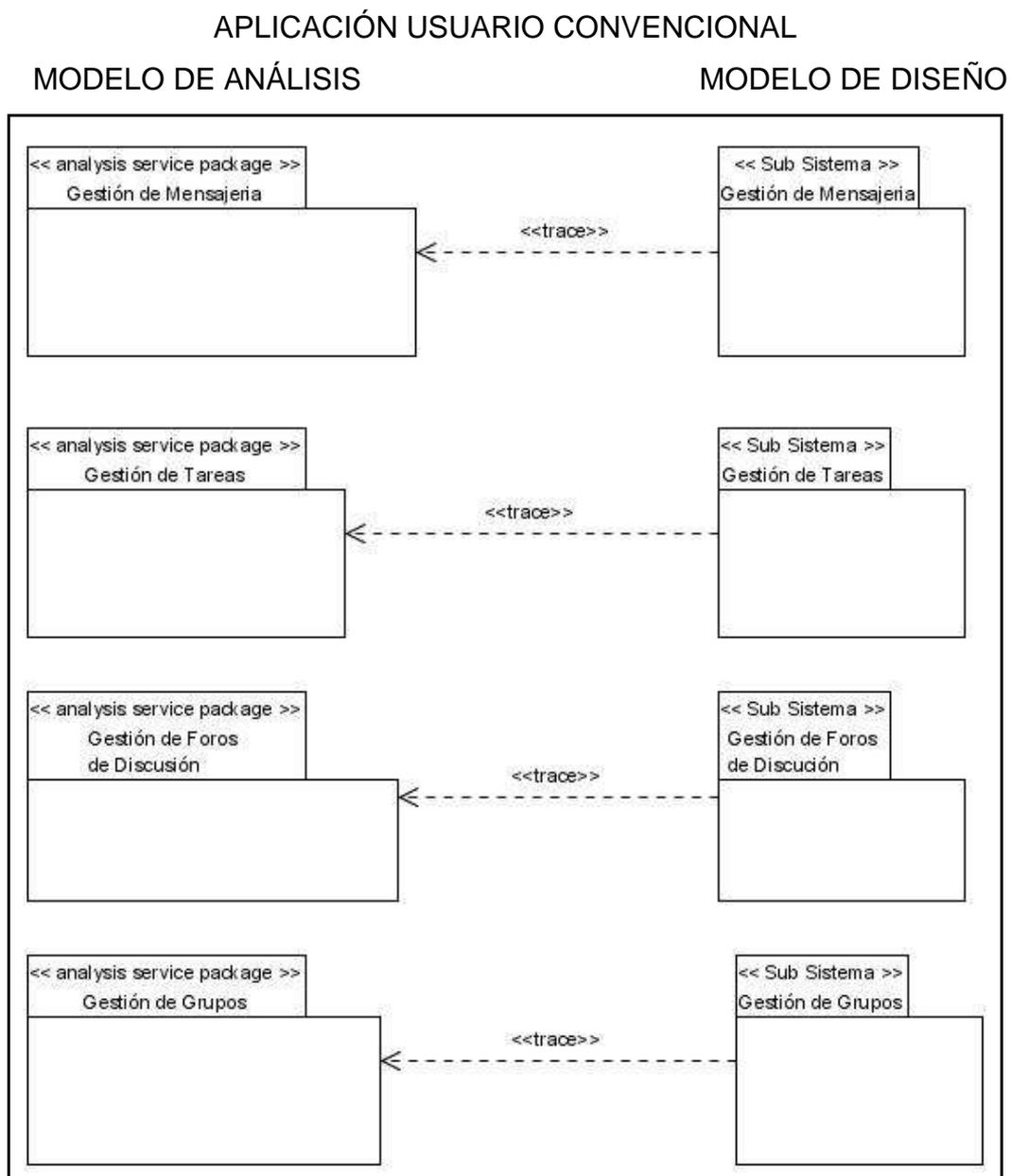
[Figura 2.72]

Elaborado por: El Autor

Referencia: Modelos desarrollados por el autor con la herramienta Poseidón.

2.4.3.- ESPECIFICACIÓN DE PAQUETES DE DISEÑO PARA LA APLICACIÓN

En la Figura 2.73 y en la Figura 2.74 se muestra la trazabilidad entre los modelos de diseño con el modelo de análisis.



[Figura 2.73]

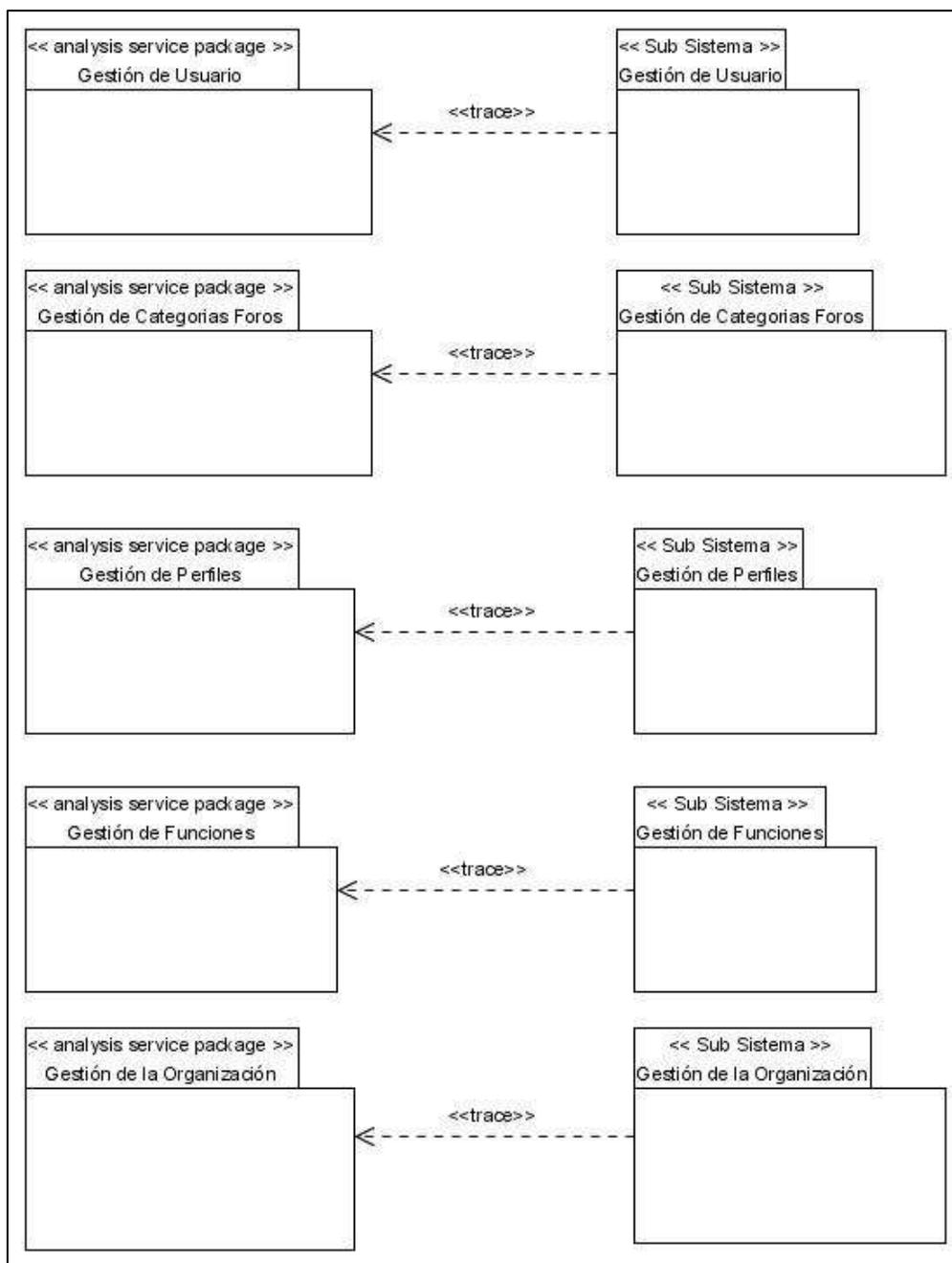
Elaborado por: El Autor

Referencia: Modelos desarrollados por el autor con la herramienta Poseidón.

APLICACIÓN USUARIO ADMINISTRADOR

MODELO DE ANÁLISIS

MODELO DE DISEÑO



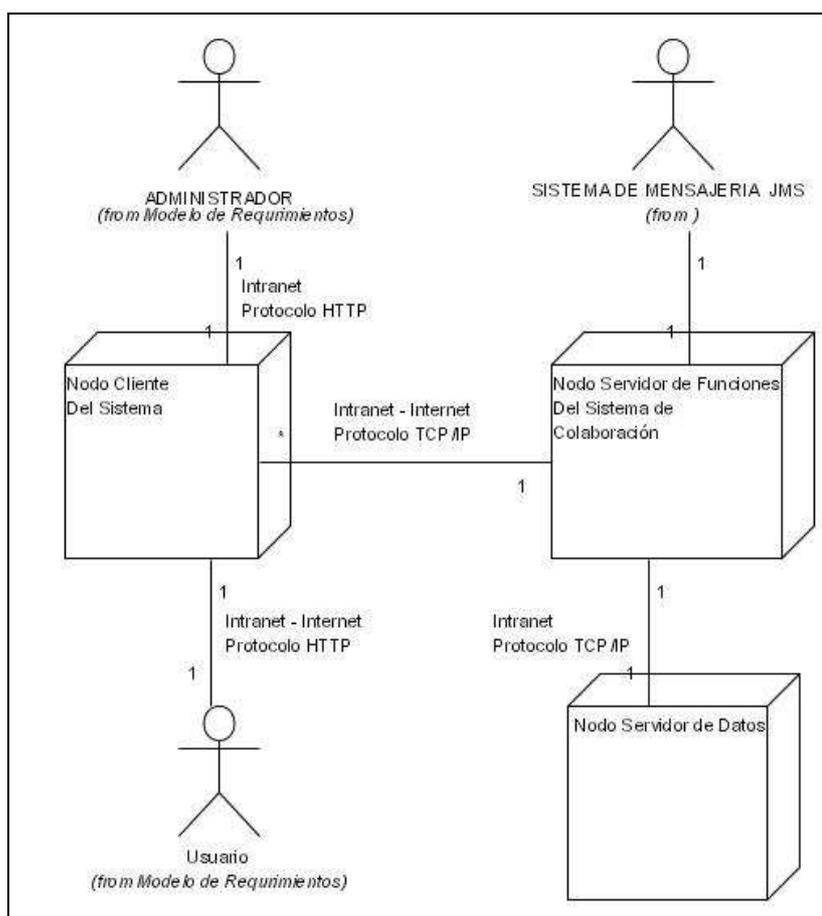
[Figura 2.74]

Elaborado por: El Autor

Referencia: Modelos desarrollados por el autor con la herramienta Poseidón.

2.4.4.- DIAGRAMA DE DESPLIEGUE PARA EL SISTEMA DE COLABORACIÓN

El Sistema de Colaboración se ejecutará sobre dos nodos servidores y un cierto número de nodos cliente. En primer lugar utilizaremos un nodo servidor que maneje y procese los objetos del negocio. En segundo lugar utilizaremos un nodo servidor de datos que maneje la información que resulte del uso del sistema. Los usuarios finales acceden al sistema mediante nodos cliente. Los nodos se comunican mediante el protocolo TCP/IP de Internet e Intranet. El nodo servidor de objetos del negocio trabajará con componentes Java. El nodo servidor de datos trabajará con una base de datos Open Source. Cada nodo cliente necesitara un explorador de páginas Web que permita la ejecución de código cliente a nivel del explorador ya sea para aplicaciones Java como para JavaScript; Véase la Figura 2.75.

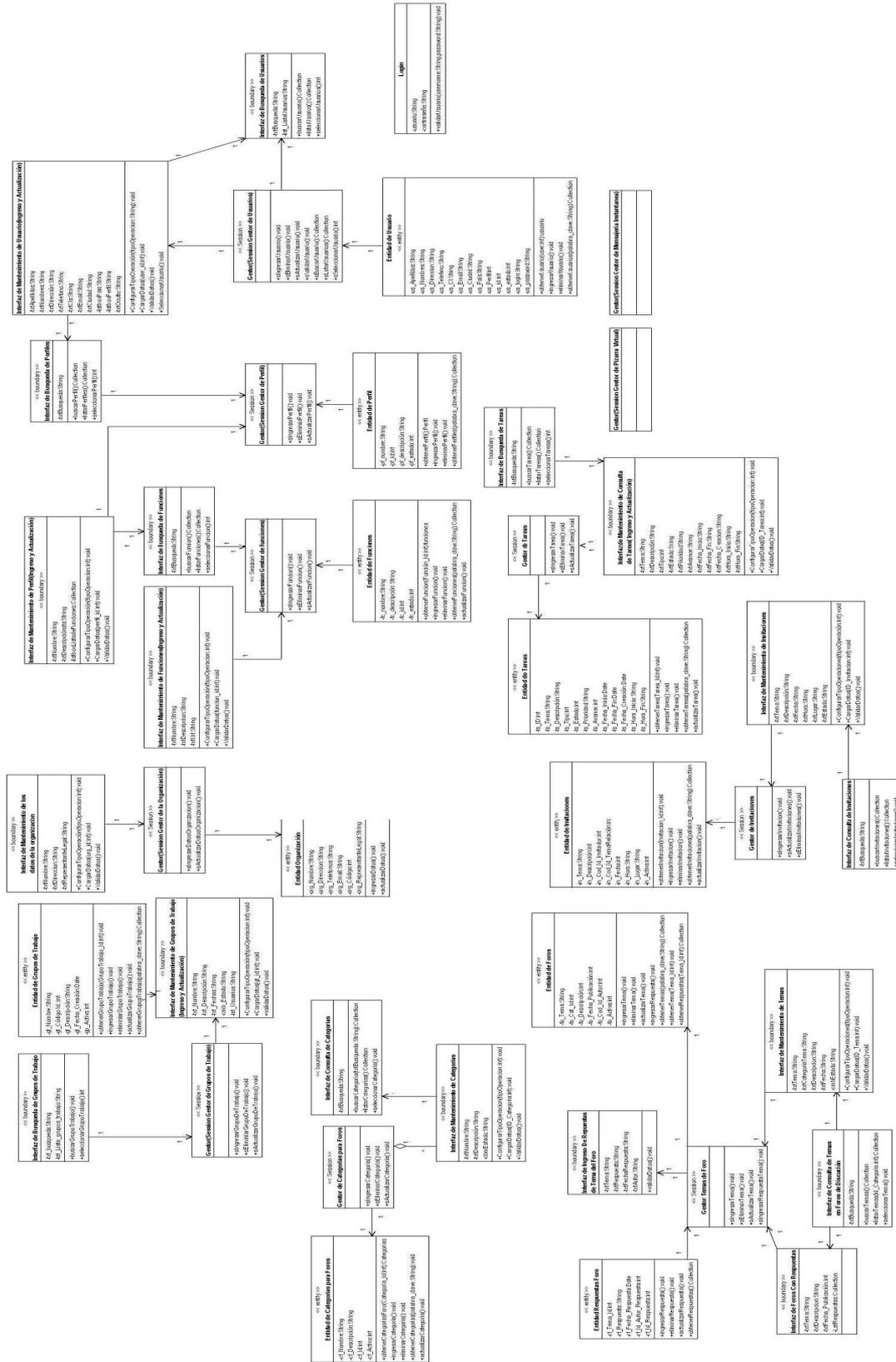


[Figura 2.75]

Elaborado por: El Autor

Referencia: Modelos desarrollados por el autor con la herramienta Poseidón.

2.4.5.- DIAGRAMA DE CLASES DE DISEÑO



[FIGURA 2.76]

Elaborado por: El Autor

Referencia: El Autor

3.- IMPLEMENTACIÓN Y PRUEBAS DEL SISTEMA

3.1.- IMPLEMENTACIÓN

3.1.1.- MODELO DE IMPLEMENTACIÓN

En la fase de implantación se procede a crear los ficheros en los cuales se pondrá el código fuente ya sean Script, Ejecutables, etc., todo esto como resultado de la fase de diseño. En el modelo de implementación se denota la implantación del sistema en términos de componentes los cuales son distribuidos en sub sistemas y a su vez son distribuidos a los nodos como se vio en el diagrama de despliegue.

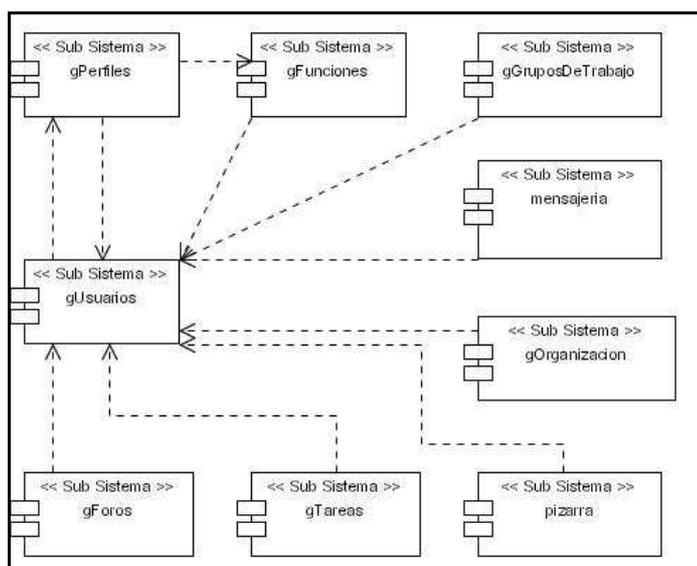
Los artefactos que se denota en la fase de implantación es el modelo de implementación el cual se compone de:

- Componentes
- Sub. Sistemas
- Interfaces
- Arquitectura del modelo de implementación.

3.1.2.- MODELO DE IMPLEMENTACIÓN PARA LA APLICACIÓN

Con el conjunto de componentes y/o sub sistemas podemos ver la relación entre los mismos, y con lo cual se define el modelo de implementación de la aplicación.

Véase Figura 3.1.



[FIGURA 3.1]

Elaborado por: El Autor

Referencia: Modelos desarrollados por el autor con la herramienta Poseidón.

Los estándares de desarrollo sobre la plataforma J2EE son los siguientes:

- Los componentes de las entidades que son clases de carácter persistente contra la base de datos empezaran con la letra “e” en minúscula seguida del nombre de la clase.
- Los componentes de las clases de sesión que manejan la lógica del negocio empezaran con la letra “s” en minúscula seguidas del nombre de la clase.
- Los componentes serializables, los applets y los servlets tendrán su nombre en minúscula.
- Los nombres de las interfaces de trabajo del usuario comenzaran con una letra minúscula seguida del nombre de la función de la página.
- Los paquetes que contienen los componentes deberán empezar con una letra “g” correspondiente a la gestión sobre las clases que son relacionadas.
- Los nombres de las tablas en la base de datos debe comenzar con la letra “e” seguida del nombre de la tabla.
- La generación de tablas de relaciones muchos a muchos debe ser realizada por el contenedor de componentes.
- Las relaciones entre tablas deben ser manejadas por el contenedor de componentes.

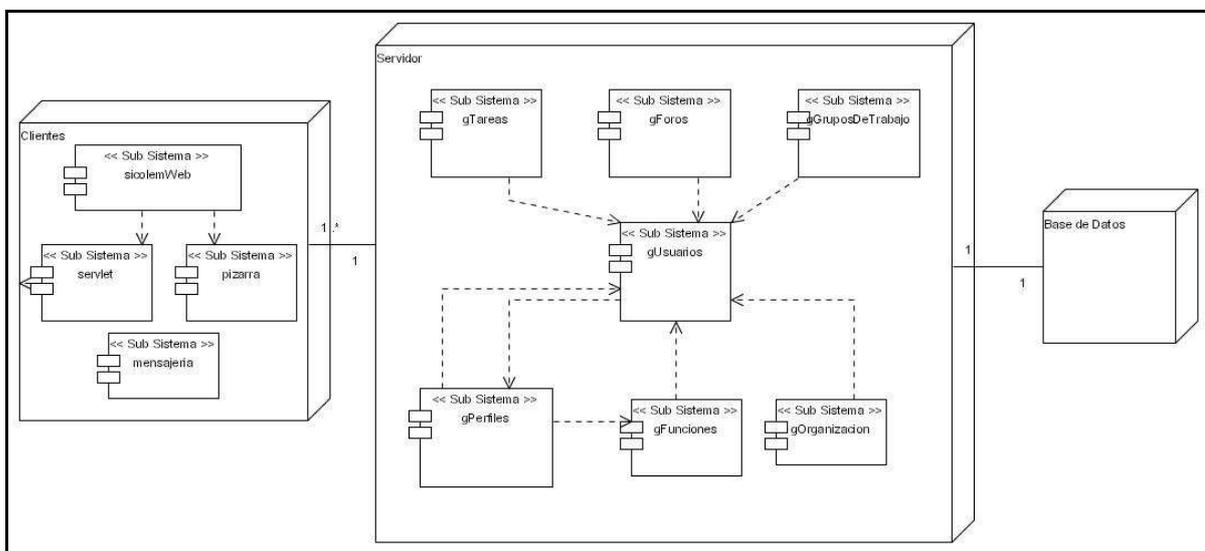
3.1.3.- DESCRIPCIÓN DE CLASES POR COMPONENTES Y/O SUBSISTEMAS

Sub Sistemas	CLASES(Componentes)
gUsuarios	sGestionUsuarioBean
	eUsuarioBean
	usuario
gPerfil	sGestionPerfilBean
	ePerfilBean
	perfil
gFunciones	eFuncionesBean
	sGestionFuncionesBean

	funciones
gOrganizacion	sGestionOrganizacionBean
	eOrganizacionBean
	organizacion
gTareas	sGestionTareaBean
	eTareaBean
	sGestionForosBean
	eCategoriasForosBean
	eForosBean
	eRespuestasForoBean
	sGestionCategoriaForoBean
	sGestionForosBean
	tarea
gGruposDeTrabajo	eGruposDeTrabajoBean
	sGestionGruposDeTrabajoBean
	gruposdetrabajo
mensajería	cMensajes
pizarra	cPizarra
	cDibujarControles
	cDibujarPanel
	cDibujarColores
	cEnvioMensaje
	cContenidoPizarra
servlet	executeu
	executep
	executef
	executeo
	executet
	executecf
	executef
	executerf

	executegt
sicolemWeb	uBusqueda
	uMantenimiento
	pBusquda
	pMantenimiento
	fBusqueda
	fMantenimiento
	oMantenimiento
	tBusqueda
	tMantenimiento
	foroBusqueda
	foroMantenimiento
	gtBusqueda
	gtMantenimiento
	mMensajería
mPizarra	

3.1.4.- AMBIENTE DE DESARROLLO



[FIGURA 3.2]

Elaborado por: El Autor

Referencia: Modelos desarrollados por el autor con la herramienta Poseidón.

De acuerdo al modelo de despliegue por componentes, Véase Figura 3.2, para efectuar la implantación es indispensable la utilización de los siguientes requisitos para el desarrollo, implantación y pruebas del sistema.

Servidor de Componentes

Instalar sistema operativo con maquina virtual de JAVA para permitir la ejecución de las clases.

Instalar un servidor de aplicaciones empresariales JBoss versión mínima 3.2.1.

Servidor Web

Instalar sistema operativo con maquina virtual de JAVA para permitir la ejecución de las clases.

Instalar servidor de aplicaciones Web para compilación del lenguaje de programación JAVA, SERVLET's y JSP's, el cual puede ser Apache Tomcat Vs. 4.1.3 o 5.

Servidor de Base de Datos

Instalar servidor de base de datos MySQL Server.

Aplicación Cliente

Instalar compilador de applet's para el navegador.

Instalar Navegador compatible con JavaScript sea Microsoft Internet Explorer 6 o superior, Firefox.

El código fuente de las clases o componentes se incluirá como un anexo en el formato digital adjunto al proyecto.

3.1.5.- PRUEBAS DEL SISTEMA

En esta fase, procederemos a realizar pruebas sobre los módulos implementados del sistema de Colaboración.

3.1.6.- MODELO DE PRUEBAS

El modelo de pruebas consta de artefactos los cuales nos ayudarán a evaluar el Sistema de Colaboración: Caso de prueba, Procedimiento de prueba, Componente de prueba, Plan de prueba, Defecto si se encuentra un problema y la Evaluación de la prueba.

Estos artefactos se unificarán para realizar las respectivas pruebas.

Plantilla de casos de pruebas.

PRUEBA N.-		CASO DE USO BASE:	
ENTRADA			
RESULTADO			
CONDICIONES			
PROCEDIMIENTO DE PRUEBA			
PLAN DE PRUEBA			
DEFECTO			
EVALUACIÓN			

[FIGURA 3.3]

Elaborado por: El Autor

Referencia: Plantilla de especificación de casos de prueba..

4.- CASO DE ESTUDIO

4.1.- DEFINICIÓN DEL CASO DE ESTUDIO

Se escogió como caso de uso una organización que necesita entablar comunicación entre varios puntos físicos distintos y que necesite de una herramienta de colaboración.

El presente proyecto se instala para comprobar su funcionalidad en un ambiente real ya que la organización seleccionada tiene como participantes a los técnicos que se comunican diariamente en distintos puntos geográficos no solo del país si no a nivel internacional.

El objetivo del caso de estudio es aplicar el sistema de colaboración en una empresa Florícola que haga uso de sus funcionalidades para que el técnico gerente de finca realice citas, tareas y conversaciones vía Chat con el gerente general de la empresa, físicamente distantes y a su vez el gerente general de la organización con los accionistas de la organización en distintas partes de mundo, por ser una aplicación Web.

La florícola es una organización enfocada en la producción y exportación de las mejores rosas de Ecuador para un mercado exclusivo brindando un servicio de preventa y postventa de la más alta calidad, porque responden con flexibilidad a las especificaciones y requerimientos de sus clientes brindando la mejor solución en el área logística de exportaciones.

La finca se ubica en un sector excepcional para la producción de rosas de alta calidad por factores como la altitud, el sol, las condiciones atmosféricas y al menos 12 horas de luz que reciben diariamente.

Los estrictos sistemas de control de calidad en cada punto del proceso y el alto grado de responsabilidad con un grupo de trabajadores responsables que se constituyen en la fuerza de trabajo, aseguran un producto que satisface altos estándares de calidad y sobre todo los requerimientos del mercado internacional.

4.2.- INSTALACIÓN DEL CASO DE ESTUDIO

Para la instalación del caso de estudio se utiliza las clases, paquetes o componentes desarrollados en el capítulo 3, se utiliza como servidor de componentes al servidor JBoss Versión 3.2.1, para la interacción con el usuario se utiliza un servidor de aplicaciones web como Apache Tomcat Versión 4.1.3 y para el almacenamiento de los datos se usa el servidor de datos MySQL, creando una estructura cliente servidor la cual es óptima para el uso en la Intranet e Internet.

Para la implantación del sistema en las estaciones de los clientes se utiliza un navegador de páginas Web el cual tenga instalado el plugin para la ejecución de applets en la página Web. El código de las páginas Web está escrito en lenguaje Java con la extensión jsp que puede ser compilada por el servidor de aplicaciones Apache Tomcat.

La infraestructura física que alojara a los respectivos servidores: de componentes, de aplicación web y de base de datos será un equipo servidor con las siguientes características físicas:

Servidor con procesador Pentium 4

Memoria de 512 MB de RAM

Disco duro Serial ATA con mínimo de 1 GB de espacio libre.

Interfase de Red local con salida a Internet.

4.3.- INGRESO DE INFORMACIÓN AL SISTEMA

4.3.1.- CARGA DE DATOS

Primero, el usuario administrador deberá configurar las funciones que el sistema tiene y crear un conjunto de perfiles con las diversas funciones escogidas, y de acceso con lo cual los usuarios tendrán derechos de uso de las funcionalidades del sistema.

Para que el sistema empiece a trabajar necesita del ingreso de usuarios y de los datos de la organización.

Para el ingreso de usuarios al sistema se requiere de una ficha personal en la que se encuentre los datos del usuario, también se debe especificar que perfil tendrá a su cargo ya que de esto depende que el usuario ingrese a ciertas funciones del sistema.

Para el ingreso de los datos de la organización se requiere de una ficha en la que consten los datos de la organización.

El ingreso de los datos se realizará por parte de un usuario administrador el cual tiene privilegios para hacer este ingreso, desde cualquier equipo que tenga acceso al sistema vía Web, esto quiere decir que el sistema estará en línea.

Con el proceso de ingreso de usuarios al sistema, se procede a informar a los usuarios para que ingresen al sistema, en este caso de estudio se informa al gerente de finca, al gerente general y a los accionistas cómo trabajar con el sistema y como ingresar al mismo.

4.3.2.- EVALUACIÓN

Para la realización de las pruebas de evaluación es necesario que el sistema esté totalmente integrado.

En esta sección se muestra a manera de ejemplo la prueba de integración que se realizó con la Interfaz de "Ingreso de Usuarios". Véase Figura 4.1. Pruebas de

validación, Pruebas del sistema, pruebas de concurrencia. El resto de pruebas se muestran en los anexos.

The screenshot shows a web browser window displaying a user registration form. At the top left is the 'Ponte Tresa' logo and 'Inversiones Ponte Tresa Cia. Ltda.'. At the top right, a user profile box shows 'Nombre: Santiago Alejandro Vaca Carrillo', 'Email: svaca@pontetresa.com', and 'Perfil: Administrator', with a 'Salir del Sistema' button. The main heading is 'INGRESO DE USUARIOS AL SISTEMA'. The form fields are: NOMBRES: Laura Elizabeth; APELLIDOS: Almeida Cardenas; CI: 1709008088; TELEFONO: 098816922; DIRECCIÓN: San Jose de Moran; EMAIL: ealmeida@pontetresa.com; PAIS: Ecuador; CIUDAD: Quito; PERFIL: ADMINISTRATOR (dropdown); ESTADO: ACTIVO(A) / INACTIVO(A); LOGIN: ealmeida; CONTRASEÑA: *****. Buttons at the bottom are 'Ingresar Información' and 'Borrar Todo'. A sidebar on the left has a menu with 'USUARIOS', 'FUNCIONES', 'PERFILES', 'CATEGORIAS DEL FORO', 'MENSAJERIA', and 'ORGANIZACION'. A calendar for September 2006 is also visible.

[FIGURA 4.1]

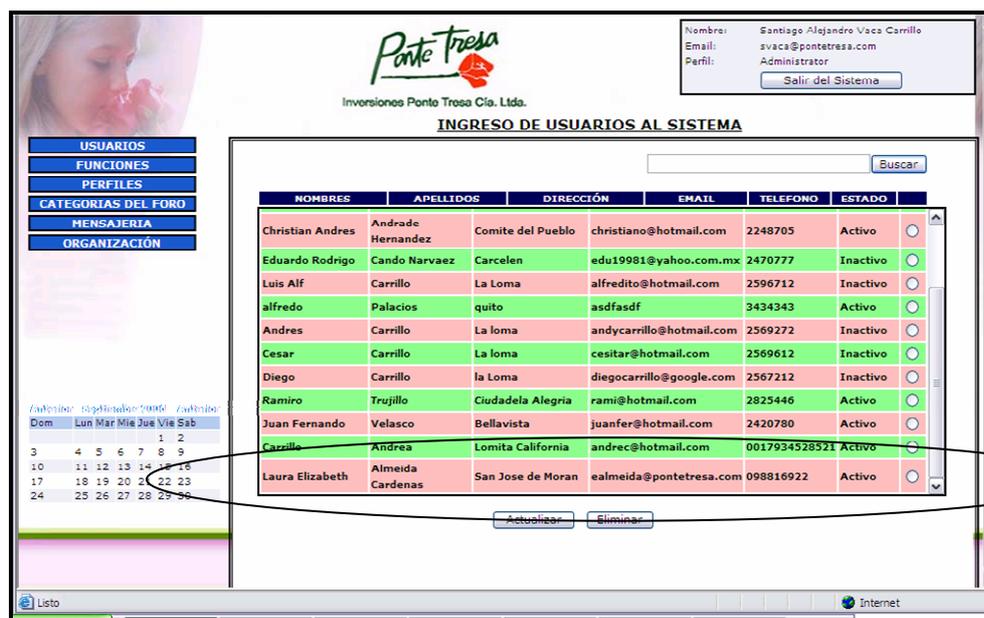
Elaborado por: El Autor

Referencia: El Autor.

En la Figura 4.1 se muestra el formulario que tiene que ser llenado por el administrador para el ingreso de los usuarios al sistema.

En la prueba de integración cuando se seleccionó en el menú en la sección de usuarios el ingreso de los usuarios, se desplegó en el área de trabajo la pantalla que contiene el formulario de ingreso de usuarios.

Posterior a ingresar al usuario se despliega un listado en el que se muestra al usuario ya en el listado.



[FIGURA 4.2]

Elaborado por: El Autor

Referencia: El Autor.

En la Figura 4.2 puede apreciar que el proceso de ingreso de usuarios esta integrado y en funcionamiento ya que devuelve la información requerida para comprobar el ingreso.

4.4.- ANÁLISIS DE RESULTADOS

Casos de Prueba del ejemplo de ingreso de un nuevo usuario al sistema.

PRUEBA N.-	1	CASO DE USO BASE:	Gestión de Usuarios
ENTRADA	Ninguna.		
RESULTADO	Listado de Usuarios del Sistema.		
CONDICIONES	Seleccionar la pantalla de Búsqueda de Usuarios únicamente si se tiene permiso para ingresar a esta pantalla.		
PROCEDIMIENTO DE PRUEBA	En el menú seleccione el botón de usuarios, se abre una pestaña que contiene el sub. menú de usuarios. En el sub. menú de usuarios seleccione búsqueda de usuarios. Se abre en el panel derecho la ventana de Usuarios con un listado general de los usuarios ingresados.		

PLAN DE PRUEBA	Esta prueba debe efectuarse con un usuario con permisos ya sea el administrador o con un perfil similar, también debe ser hecha para comprobar la funcionalidad del listado por filtro por alguna condición en particular.
DEFECTO	Ninguno
EVALUACIÓN	INGRESO A PANTALLA CORRECTO LISTADO PRINCIPAL CORRECTO LISTADO POR FILTRO CORRECTO

PRUEBA N.-	2	CASO DE USO BASE:	Gestión de Usuarios
ENTRADA	Ninguna.		
RESULTADO	Ingreso de usuarios.		
CONDICIONES	<p>Seleccionar la pantalla de Ingreso de Usuarios únicamente si se tiene permiso para ingresar a esta pantalla.</p> <p>Llenar los campos que sean obligatorios.</p> <p>Llenar los campos con valores reales.</p>		
PROCEDIMIENTO DE PRUEBA	<p>En el menú seleccione el botón de usuarios, se abre una pestaña que contiene el sub. menú de usuarios.</p> <p>En el sub. menú de usuarios seleccione ingreso de usuarios. Se abre en el panel derecho la ventana de Usuarios con una forma de ingreso que se encuentra vacía.</p> <p>Ingreso en los campos vacíos y selección de campos con variables de opciones predefinidas.</p> <p>Al final de la forma seleccione el botón de ingreso.</p>		
PLAN DE PRUEBA	Esta prueba debe efectuarse con un usuario con permisos ya sea el administrador o con un perfil similar, también debe ser hecha para comprobar la funcionalidad del ingreso y registro de un nuevo usuario.		
DEFECTO	Si la pantalla después del ingreso no muestra el listado y muestra la misma pantalla deberá ser por que se requiere corregir los datos.		

EVALUACIÓN	Ingreso a pantalla CORRECTO
	Ingreso de registro CORRECTO
	Devolución para corrección CORRECTO

Con la información ingresada en el formulario de pruebas, sacamos como conclusión que el proceso ejemplo se efectúa correctamente utilizando un equipo de características mínimas mostrando un alto grado de respuesta oportuna y correcta.

5.- CONCLUSIONES Y RECOMENDACIONES

5.1.- CONCLUSIONES

El Sistema de Colaboración proporciona una visión amplia de las necesidades que una organización puede tener al momento de discutir, opinar, compartir, comentar y apoyarse, en uno o varios temas, lo cual es importante al momento de tomar decisiones.

Se concluye que el desarrollo comunicacional en un organización es muy importante, y este proyecto se enfocó en brindar el apoyo al mismo y tratar de explotar las potencialidades de comunicación entre los usuarios.

Las herramientas open source benefician el entendimiento y desarrollo de las habilidades de la persona que implementa soluciones automatizadas.

La plataforma J2EE en desarrollo tiene herramientas que son muy completas como JBoss que fue de gran apoyo para la realización de este proyecto.

Las especificación JMS en un entorno de J2EE es muy amplia y fácil de manejar como resultado ayudó para la comunicación asincrónica que tiene el proyecto con lo cual sus respectivas funciones tuvieron un gran impacto satisfactorio en el mismo.

El API de JMS permite que sea fácil la integración con otras herramientas de mensajería, con lo cual se permite que sea factible trabajar con otras herramientas que se acoplen a la especificación JMS.

Se concluye que es importante entender los procesos que comprenden el ciclo de vida del software ya que es la base para poder seleccionar correctamente una metodología de desarrollo.

El modelo en espiral brinda la capacidad de iterar lo cual es beneficioso al momento de mejorar o corregir un proceso.

Se concluye que los modelos de casos de uso son necesarios para analizar las necesidades funcionalidades del sistema y aclaran la forma de proceder con el desarrollo del sistema.

Los modelos de casos de uso participaron activamente en todo el proceso de desarrollo del sistema de software, lo cual es producto de la relación que tienen los modelos que están implementados, esta relación es llamada trazabilidad y a través de las iteraciones y la trazabilidad se obtuvo mejores modelos y soluciones a las necesidades del sistema.

El proceso unificado de desarrollo del software es una metodología que cumple con las necesidades de desarrollo ya que incorpora el modelo en espiral y organiza correctamente los procesos del ciclo de vida.

La metodología aplicada para desarrollar el sistema hace que sea un proyecto no de corto alcance, con lo cual se debe mejorar el sistema en un futuro para que el factor comunicacional alcanzado no se desaproveche y mejor se analice la oportunidad de aportar con más funcionalidades al mismo.

La orientación a objetos tiene y tendrá un enfoque muy práctico para solucionar problemas ya que ayudó en la organización del desarrollo y ejecución del proyecto.

La herramienta Poseidón en la etapa de análisis no ayudo a generar los modelos de colaboración como esta especificado en el lenguaje UML, por lo que se procedió a utilizar las opciones que se permitían.

5.2.- RECOMENDACIONES

Para la selección de una metodología, se recomienda analizar el ciclo de vida de desarrollo del software y un modelo de trabajo que se acople a las necesidades del sistema.

No se pretende que otras personas sigan los pasos que se utilizaron para el desarrollo o escojan la metodología que se utilizó en este proyecto, pero si se recomienda que tomen a este proyecto como una referencia ya que se realizo una investigación para poder clarificar como seleccionar una metodología que se acople a las necesidades del sistema, esto puede ser una fuente de apoyo para otros proyectos.

Se recomienda utilizar el lenguaje de programación JAVA cuando se necesite de un lenguaje orientado a objetos.

Se recomienda utilizar sistemas de código abierto ya que son una fuente de aprendizaje y mejoramiento en la calidad de desarrollo por parte de los participantes del proyecto.

Se recomienda utilizar la licencia GPL para los sistemas desarrollados sobre herramientas de código abierto ya que la misma permite que se modifiquen los programas para su mejoramiento y a la vez se distribuya y comparta el conocimiento.

Se recomienda el uso de la arquitectura de desarrollo J2EE ya que trabaja en n capas y separa los módulos desarrollados en componentes, lo cual mejora el desarrollo y funcionamiento del sistema.

Se recomienda el uso de la especificación JMS para proyectos de mensajería ya permite una manera fácil de comunicación.

Se recomienda utilizar UML como lenguaje de modelado de objetos ya que es una fuente de clarificación sobre las necesidades que el proyecto tiene.

Se recomienda no utilizar la herramienta Poseidón para diagramar con el lenguaje UML ya que tiene muchas complicaciones para la generación de diagramas necesarios para el cumplimiento de la metodología, es preferible buscar una herramienta que brinde el apoyo para la elaboración de diagramas con UML.

BIBLIOGRAFÍA

- [1] AMBLER Scott, CONSTANTINE Larry, The Unified Process: The Transition and Production Phases, CMP Books, Kansas, 2002
- [2] GARCIA Javier, RODRIGUEZ José Ignacio, MINGO Iñigo, IMAZ Aitor, BRAZALEZ Alfonso, LARZABAL Alberto, CALLEJA Jesús, GARCIA Jon, Aprende JAVA, Escuela Superior de Ingenieros Industriales, San Sebastián, 1999
- [3] JACOBSON Ivar, BOOCH Grady, RUMBAUGH James, El Proceso Unificado de Desarrollo de Software, Primera Edición, Pearson Education, Madrid, 2000
- [4] LALANI Suleiman, JAMSA Kris, Biblioteca del Programador JAVA, Primera Edición, McGRAW HILL INTERAMERICANA EDITORES S.A., 1997
- [5] PRESSMAN Roger S., Ingeniería de Software Un Enfoque Práctico, Quinta Edición, McGraw Hill, 2002
- [6] ARMELINI Guillermo, Nuevos retos para la mensajería instantánea, URL: www.iese.edu/es/files/5_7838.pdf, 2003
- [7] BARRIENTOS ENRÍQUEZ Aleida Mirian, El desarrollo de sistemas de información empleando el lenguaje de modelado unificado UML, URL: <http://www.monografias.com/trabajos16/lenguaje-modelado-unificado/lenguaje-modelado-unificado.shtml>, 2003
- [8] CASTRO GIL Robin Alberto, Estructura básica del proceso unificado de desarrollo de software, URL:

- http://www.icesi.edu.co/es/publicaciones/publicaciones/contenidos/sistemas_telematica/3/rcaastro_estructura-bas-puds.pdf, Universidad Icesi, 2004.
- [9] "Categorías de Software Libre y No Libre", URL:
<Http://www.gnu.org/philosophy/categories.es.html#FreeSoftware> Copyright (C) 1996, 1997, 1998 Free Software Foundation, Inc., 51 Franklin St, Fifth Floor, Boston, MA 02110, USA
- [10] FERNÁNDEZ SÁNCHEZ José Luis, Reusabilidad y Desarrollo Orientado a Objetos, URL: <http://www.ati.es/gt/LATIGOO/OOp96/Ponen7/atio6p07.html>, 1998.
- [11] "FREE SOFTWARE FUNDATION", GNU General Public License, URL:
<http://www.fsf.org/licensing/licenses/gpl.html>, 2005
- [12] MENDOZA Sánchez María, Metodologías De Desarrollo De Software, URL:
<http://www.informatizate.net>, 07/06/2004
- [13] MEIRA Carlos María, PARÓN Angeles, ROSANO Déborah, Las tecnologías de colaboración serán decisivas para el desarrollo de la industria en los próximos tres años, URL:
http://about.reuters.com/latam/prensa/prensa_gartner.asp, 2003.
- [14] MOLPECERES Alberto, PEREZ Martín, Arquitectura Empresaria y Software Libre, J2EE, Segunda Revisión, URL: <http://www.javahispano.org>, 2002.
- [15] NICOL Chris, LA ASOCIACIÓN PARA EL PROGRESO DE LAS COMUNICACIONES, URL: http://www.egipuzkoa.net/jardunaldiak/docs/castellano/jornadas_anteriores/chris_nicol_es.pdf, 2004.
- [16] SÁNCHEZ PÉREZ Javier, Metodologías de Desarrollo Software- Proceso Unificado de Desarrollo de Software, URL:
<http://www.infor.uva.es/~mlaguna/is2/2-6-Proceso.pdf>, 2003.

ANEXOS

Los anexos se adjuntarán en la unidad óptica del proyecto de titulación e incluyen:

ANEXO 1

- **MANUAL DE INSTALACIÓN**

ANEXO 2

- **MANUAL DE USUARIO**

ANEXO 3

- **CÓDIGO FUENTE**

ANEXO 4

- **PRUEBAS DEL SISTEMA**

ANEXO 5

- **INSTALADORES**

ANEXO 6

- **DOCUMENTO TÉCNICO**

ANEXO 7

- **ARCHIVO DE LOS MODELOS DE LA HERRAMIENTA POSEIDÓN**