



Universidad de las Ciencias Informáticas

Facultad 4

Diseño e implementación de un módulo para la integración de Moodle con un sistema de mensajería instantánea.

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

Autor:

Reinier Alcayde Pupo

Tutor(es):

Ing. Juenlis E. Coss Piña

Ing. Jesús Hidalgo Guillén

Declaración de autoría

Declaro ser autor de la presente tesis y reconozco a la Universidad de las Ciencias Informáticas los derechos patrimoniales de la misma, con carácter exclusivo.

Para que así conste firmamos la presente a los ___días del mes de ____del año ____.

Reinier Alcayde Pupo

Firma del autor

Ing. Juenlis Enrique Coss Piña

Firma del Tutor

Ing. Jesús Hidalgo Guillén

Firma del Tutor

Resumen

Con el desarrollo de la sociedad, la educación no se ve ajena al uso de las nuevas tecnologías y su aplicación en el proceso de enseñanza–aprendizaje propició el surgimiento del e-learning. Este, ha tomado un papel protagónico en la educación, debido a que integra un grupo de tecnologías que facilitan y enriquecen la interrelación entre alumnos y profesores mediante plataformas tecnológicas, que contienen herramientas que permiten la comunicación a través de intranet e internet. Dentro de estas plataformas se encuentra Moodle, la cual es ampliamente utilizada para apoyar el proceso de enseñanza-aprendizaje. En la Universidad de las Ciencias Informáticas (UCI), su aplicación alcanza grandes resultados en el proceso de enseñanza-aprendizaje. Aunque se han obtenido buenos resultados con la utilización de Moodle en la universidad, aún se pueden encontrar deficiencias en la comunicación síncrona que afectan la calidad y eficiencia de los procesos que se llevan a cabo dentro de la plataforma. Para dar solución a esta problemática se desarrolló un módulo de chat sobre el protocolo XMPP, que utilizando tecnología AJAX permite que los profesores puedan crear salas de chat para debatir temas relacionados con las asignaturas, donde los estudiantes puedan resolver cualquier duda relacionada con el debate.

Palabras Clave: comunicación síncrona, salas de chat, jabber/XMPP, e-learning, Moodle.

Índice

Introducción	1
Capítulo 1: Fundamentación Teórica	5
Introducción	5
1.1 E-learning	5
1.2 Sistema para la Administración de Aprendizaje	6
1.2.1 La plataforma de teleformación Moodle	7
1.3 La Mensajería Instantánea	7
1.3.1 Moodle y la Mensajería Instantánea	8
1.3.1.1 Módulos de Moodle para Mensajería Instantánea	9
1.3.2 La Mensajería Instantánea en otros LMS	11
1.4 Tecnologías para la implementación de sistemas de mensajería instantánea ...	12
1.4.1 Protocolos para Mensajería Instantánea	12
1.4.2 Sistemas y librerías para la implementación de servicios de mensajería instantánea.....	16
1.5 Lenguajes de Programación para implementar sistemas de mensajería instantánea.....	19
1.6 Herramientas para implementar sistemas de mensajería instantánea basados en web.....	23
1.7 Proceso de desarrollo de software.....	24
1.7.1 Metodología de Desarrollo de Software	24
1.7.2 Lenguaje Unificado de Modelado	27
1.8 Herramientas CASE para el modelado	27
1.8.1 Rational Rose Enterprise Edition	27
1.8.2 Visual Paradigm for UML.....	28
Conclusiones parciales.....	28
Capítulo 2: Análisis y Diseño de un módulo para la integración de Moodle con un sistema de mensajería instantánea.	30
Introducción	30
2.1 Propuesta del sistema	30
2.2 Modelo de Dominio	30

Índice

2.3 Definición de Requerimientos Funcionales	31
2.4 Definición de Requerimientos No Funcionales	32
2.5. Definición de los Actores del Sistema.....	33
2.6 Diagrama de Casos de Uso del Sistema	34
2.7 Descripción Textual de los Casos de Uso	34
2.8 Modelo de Análisis.....	45
2.9 Diagramas de Interacción.....	47
2.10 Modelo de Diseño	52
2.11 Modelo de Datos	55
Conclusiones parciales.....	56
Capítulo 3: Implementación del Módulo ChatXMPP	57
Introducción	57
3.1 Modelo de Implementación	57
3.2 Pruebas	59
Conclusiones Generales	70
Recomendaciones	71
Bibliografía Referenciada	72
Glosario de Términos	75
Anexos	76

Introducción

El uso de las Tecnologías de la Información y las Comunicaciones (TIC), desempeñan un papel importante en el desarrollo de muchos sectores de la sociedad. En la educación llegaron con gran impacto, trayendo consigo numerosas ventajas para los sistemas educativos tradicionales. La aplicación de dichas tecnologías en este sector propició el surgimiento de una nueva modalidad educativa, la Educación a Distancia; demostrando que es posible aprender sin estar reunidos en el mismo lugar y al mismo tiempo.

El e-learning es un sistema que integra un grupo de tecnologías que facilitan y enriquecen la interrelación entre alumnos y profesores mediante plataformas tecnológicas que contienen herramientas que permiten la comunicación a través de intranet e internet.

Un lugar relevante en el mundo del e-learning lo ocupan los LMS (Sistema de Gestión del Aprendizaje), sistemas que permiten, además, de crear y administrar los contenidos, controlar el proceso de enseñanza-aprendizaje. Entre los LMS más difundidos se encuentra Moodle (Entorno de Aprendizaje Dinámico Orientado a Objetos y Modular), plataforma gratuita de gran flexibilidad que permite la creación de cursos en línea. Moodle, además, que es de código abierto, permite satisfacer las necesidades de miles de estudiantes y profesores, permitiéndoles implementar estrategias de enseñanza-aprendizaje complementarias a las clases presenciales y diseñar modelos totalmente a distancia.

En diferentes universidades cubanas se emplean de manera satisfactoria los LMS, obteniéndose buenos resultados en el proceso de enseñanza aprendizaje. Sin estar ajena al desarrollo del proceso educacional, la Universidad de las Ciencias Informáticas (UCI) comienza a utilizar en el año 2005 la plataforma Moodle para dar seguimiento al proceso de enseñanza-aprendizaje, adquiriendo importancia para el desarrollo educativo, permitiendo a los estudiantes encontrar orientaciones, ejercicios de autoestudio y tareas extraclases.

Por otra parte, Moodle permite la creación de cursos y actividades en línea, así como la comunicación síncrona y asíncrona entre los usuarios. Los módulos de actividad son ejemplo de esta comunicación; foros, wikis y chats son utilizados por los usuarios para intercambiar información y debatir sobre cualquier tema. Además, permite a los usuarios crear un ambiente de colaboración haciendo ameno el aprendizaje dentro de la plataforma.

Introducción

Dentro de Moodle, para mantener una conversación sincrónica entre estudiantes y profesores se hace uso del módulo chat. Los usuarios dentro de éste tienen la posibilidad de compartir experiencias e ideas, desarrollar habilidades cognoscitivas y enriquecerse culturalmente. Esta actividad, posee varias características que lo hacen favorable para el proceso de formación que brinda Moodle; por ejemplo: soporta direcciones URL, emoticonos, integración de HTML, imágenes, y otras. Además, las sesiones de los usuarios quedan registradas y pueden programarse sesiones periódicas para fijar un encuentro y conversar de un tema determinado con uno o más usuarios, estas sesiones son utilizadas por los profesores para tener un encuentro con sus estudiantes periódicamente.

Sin embargo, el chat de la plataforma Moodle presenta varias dificultades que impiden al proceso de enseñanza desarrollarse con la eficiencia requerida; el alto consumo de recursos que causa esta actividad reduce considerablemente el rendimiento de la misma, limitando sensiblemente la calidad en las comunicaciones, donde muchos de los usuarios no reciben los mensajes de otros, y en ocasiones son desconectados.

Otro de los problemas presentes es que su interfaz, presenta una programación incipiente con respecto al desarrollo actual de los chats. Además, en ocasiones los usuarios al intentar conectarse, no pueden acceder a esta actividad y en las salas de conferencias los mensajes enviados no son mostrados o no permite escribir ningún texto.

Por lo anteriormente expresado se plantea como **problema a resolver**: ¿Cómo mejorar la comunicación instantánea de los usuarios en la plataforma de teleformación Moodle?

EL **objeto de estudio** fijado en la presente investigación es las tecnologías y estándares que permiten la implementación de sistemas de mensajería instantánea.

El **objetivo general** planteado por este trabajo es desarrollar un módulo para la integración de Moodle con un sistema de mensajería instantánea.

El **campo de acción** en que se enmarca el trabajo es la implementación de sistemas de mensajería instantánea en Moodle.

Se plantea la siguiente **idea a defender**: Una mejor comunicación instantánea de los usuarios en la plataforma Moodle es posible a través de su integración con un sistema de mensajería instantánea.

De este objetivo se desglosan los siguientes **objetivos específicos**:

Introducción

- Elaborar el marco conceptual de la mensajería instantánea.
- Realizar el análisis y diseño del módulo de mensajería instantánea.
- Implementar el módulo de mensajería instantánea.
- Realizar las pruebas al módulo de mensajería instantánea.

Para dar cumplimiento a los objetivos específicos se planifican las siguientes tareas:

- Análisis de conceptos, tecnologías y tendencias más utilizadas en el campo del e-Learning.
- Estudio del funcionamiento del módulo chat de Moodle.
- Estudio de las soluciones existentes para la integración de aplicaciones web con clientes de mensajería instantánea.
- Estudio de la arquitectura de Moodle y el desarrollo de módulos.
- Estudio de las posibles soluciones desarrolladas como alternativa al módulo chat de Moodle.
- Realización del análisis y diseño de un módulo para integrar Moodle con un sistema de mensajería instantánea.
- Implementación de un módulo para la integración de Moodle con un sistema de mensajería instantánea.
- Realizar pruebas de software.

Métodos investigativos

Para llevar a cabo la investigación, se emplean los métodos de nivel teórico y empírico.

Métodos teóricos:

- Análisis Histórico-Lógico: En esta investigación se utiliza este método para realizar el estudio del estado del arte. Es decir, investigar acerca de otras aplicaciones de mensajería instantánea así como indagar sobre los protocolos a utilizar para la mensajería instantánea, además de las herramientas de desarrollo. Describir la metodología, herramientas y lenguaje a utilizar en el análisis, diseño e implementación del módulo para la integración de Moodle con un sistema de mensajería instantánea, y estudiar la arquitectura de la plataforma Moodle.
- Modelación: El uso de este método permitirá realizar el modelo de Casos de Uso que da cumplimiento a los requisitos funcionales y no funcionales asociados al módulo para la integración de Moodle con un sistema de

Introducción

mensajería instantánea, así como realizar los diagramas de clases de diseño para cada caso de uso y diagramas de interacción aplicando patrones de diseño, el modelo de datos, la distribución física del sistema y la implementación de los componentes necesarios para dar solución al diagrama de clases de diseño correspondiente a dicho módulo.

Métodos empíricos:

- Observación: Este método es el instrumento que permite estudiar más de cerca el objeto de la investigación, las acciones, causas, consecuencias, etc. Se puede observar cómo funciona módulo Chat de Moodle y los principales problemas que tiene el mismo.

Posibles resultados:

Con el progreso de esta tesis se pretende lograr:

- Análisis y diseño de un módulo para la integración de Moodle con un sistema de mensajería instantánea.
- Implementación de un módulo para la integración de Moodle con un sistema de mensajería instantánea.
- La documentación generada constituiría una guía para futuros desarrollos sobre Moodle.

El presente trabajo está estructurado en tres capítulos:

Capítulo 1: Fundamentación Teórica. Se exponen los elementos teóricos que sustentan el problema científico y los objetivos del trabajo. Se analizan las tecnologías para la implementación de chats. Se realiza un análisis de las metodologías y herramientas de desarrollo que se pueden utilizar y se justifica la elección de cada una de ellas.

Capítulo 2: Características del Sistema. Se especifican los requisitos que debe cumplir el sistema así como el diagrama de casos de uso del sistema y las descripciones de los mismos.

Capítulo 3: Implementación del Módulo Chat XMPP. Se describen todos los artefactos generados durante la implementación del Módulo Chat de Moodle describiéndose desde el enfoque de la programación, a través del modelo de datos, los diagramas de componentes y de despliegue. Finalmente se realiza la validación de la solución propuesta.

Capítulo 1: Fundamentación Teórica

Introducción

En el presente capítulo se expondrán conceptos referentes al e-learning, centrándose fundamentalmente en la plataforma Moodle. Se analizarán las tecnologías más utilizadas para desarrollar sistemas de mensajería instantánea, además del estudio de las herramientas a utilizar y se explicarán detalladamente los aspectos a tener en cuenta en la selección de las herramientas, tecnologías y metodología a utilizar para el desarrollo de la propuesta de solución.

1.1 E-learning

Con el desarrollo de la sociedad, la educación no está ajena al uso de las nuevas tecnologías y su aplicación en el proceso de enseñanza–aprendizaje propició el surgimiento del e-learning. El e-learning proporciona la oportunidad de crear ambientes de aprendizaje, interactivos, eficientes, fácilmente accesibles y centrados en el estudiante.

Muchos autores reducen el alcance del e-learning centrándose exclusivamente en el ámbito de internet.

“... el uso de tecnologías Internet para la entrega de un amplio rango de soluciones que mejoran el conocimiento y el rendimiento. Está basado en tres criterios fundamentales:

- 1. El e-Learning trabaja en red, lo que lo hace capaz de ser instantáneamente actualizado, almacenado, recuperado, distribuido y permite compartir instrucción o información.*
- 2. Es entregado al usuario final a través del uso de ordenadores utilizando tecnología estándar de Internet.*
- 3. Se enfoca en la visión más amplia del aprendizaje que van más allá de los paradigmas tradicionales de capacitación”. (Rosenberg, 2001)*

Alejandra Souto en su definición también lo enfoca en el uso de internet *“Enseñanza a distancia caracterizada por la separación física entre profesorado y alumnado, sin excluir encuentros físicos puntuales, entre los que predomina una comunicación de doble vía asíncrona donde se usa preferentemente internet como medio de comunicación y de distribución del conocimiento, de tal manera que el alumno es el*

Capítulo 1: Fundamentación Teórica

centro de la formación independiente y flexible, al tener que gestionar su propio aprendizaje, generalmente con ayuda de tutores externos” (Souto, 2006)

El alcance del e-learning no debe acotarse solamente a internet, sino que debe adecuarse a cada entorno o situación específica. Es por eso que en esta investigación se definen como *“la capacitación no presencial que, a través de plataformas tecnológicas, posibilita y flexibiliza el acceso y el tiempo en el proceso de enseñanza-aprendizaje, adecuándolos a las habilidades, necesidades y disponibilidades de cada discente, además de garantizar ambientes de aprendizaje colaborativos mediante el uso de herramientas de comunicación síncrona y asíncrona, potenciando en suma el proceso de gestión basado en competencias” (García Peñalvo, 2005).*

El e-learning ha tomado un papel protagónico en la enseñanza debido a sus características entre las cuales se encuentran (Universidad de Sevilla, 2007):

- **“Desaparecen las barreras espacio-temporales.** *Los estudiantes pueden realizar un curso en su casa o lugar de trabajo, estando accesibles los contenidos cualquier día a cualquier hora. Pudiendo de esta forma optimizar al máximo el tiempo dedicado a la formación.*
- **Formación flexible.** *La diversidad de métodos y recursos empleados, facilita que se puedan adaptar a las características y necesidades de los estudiantes.*
- **El alumno es el centro** *de los procesos de enseñanza-aprendizaje y participa de manera activa en la construcción de sus conocimientos, teniendo capacidad para decidir el itinerario formativo más acorde con sus intereses.*
- **El profesor,** *pasa de ser un mero transmisor de contenidos a un tutor que orienta, guía, ayuda y facilita los procesos formativos.*
- **Contenidos actualizados.** *Las novedades y recursos relacionados con el tema de estudio se pueden introducir de manera rápida en los contenidos, de forma que las enseñanzas estén totalmente actualizadas.*
- **Comunicación constante** *entre los participantes, gracias a las herramientas que incorporan las plataformas e-Learning (foros, chat, correo-e, etc.).”*

El proceso de formación basado en e-learning se lleva a cabo a través de plataformas que son las que crean, gestionan y distribuyen los cursos por la web, además de permitir la interrelación entre los usuarios y los contenidos.

1.2 Sistema para la Administración de Aprendizaje

Entre los tipos de plataformas existentes dentro del e-learning encontramos los LMS (Sistema de Administración de Aprendizaje). Un LMS es un software basado en un servidor Web que provee módulos para los procesos administrativos y de seguimiento

Capítulo 1: Fundamentación Teórica

que se requieren para un sistema de enseñanza-aprendizaje, simplificando el control de estas tareas (López Guzmán, 2005).

Permiten planificar el aprendizaje de acuerdo con las necesidades de los usuarios, sean estos estudiantes, trabajadores, empresas, etc. Permiten también mejorar las competencias de los usuarios de los cursos y su intercomunicación. Es posible adaptar la formación a los requisitos de la empresa y al propio desarrollo profesional. Permiten la distribución de cursos, recursos, noticias y contenidos relacionados con la formación en general; además, pueden servir como soporte para el registro de los cursistas, acceso a recursos tales como material audiovisual, demos, etc. Entre las plataformas comerciales más comunes se encuentran Blackboard, WebCT, EduStance y de software libre Moodle, ATutor y Claroline.

1.2.1 La plataforma de teleformación Moodle

El Entorno de Aprendizaje Dinámico Modular Orientado a Objetos (MOODLE) es una plataforma para la creación de cursos y sitios web basados en Internet. Tiene como misión principal desarrollar nuevas teorías educativas basadas en toda una gama de recursos didácticos disponibles en ella.

Moodle es muy utilizado en varios centros de enseñanza, constituyendo una herramienta muy útil para estos propósitos, pues soporta disímiles contenidos como son textos, animaciones, vídeos, imágenes, entre otros. Permite fomentar el auto-aprendizaje y el aprendizaje colaborativo, realizar exámenes online y evaluar las tareas de los estudiantes.

Este sistema se distribuye gratuitamente como software libre bajo la licencia pública GNU1, puede funcionar en cualquier computador en el que pueda correr PHP, soporta varios tipos de bases de datos, esencialmente MySQL y PostgreSQL, y asegura un alto rendimiento, combinando una buena elección del hardware, sistema operativo y sistema de bases de datos.

“Moodle cuenta con una gran variedad de módulos para la creación de cursos, incluyendo recursos y actividades. Entre los que se pueden encontrar las lecciones, tareas, cuestionarios, encuestas, libros, glosarios, wikis, foros, chats, enlaces y etiquetas; con estructuras diferentes según sus objetivos específicos” (Moodle, 2010).

1.3 La Mensajería Instantánea

En la actualidad la Mensajería Instantánea juega un papel importante en la

¹ Licencia orientada a proteger la libre distribución, modificación y uso de software.

Capítulo 1: Fundamentación Teórica

comunicación. Esta se ha convertido en uno de los medios más utilizados y se define como *“Todo programa que permite dialogar en directo con un interlocutor distante mediante teclado, micrófono o teléfono”* (Anderruthy, 2007).

Una primera forma de mensajería instantánea fue la implementación en el sistema PLATO usado al principio de la década de 1970. Más tarde, el sistema talk implementado en UNIX/LINUX comenzó a ser ampliamente usado por ingenieros y académicos en las décadas de 1980 y 1990 para comunicarse a través de internet. ICQ fue el primer sistema de mensajería instantánea para ordenadores con sistema operativo distinto de UNIX/LINUX en noviembre de 1996. A partir de su aparición, un gran número de variaciones de mensajería instantánea han surgido y han sido desarrollados en paralelo en otras partes, cada aplicación teniendo su propio protocolo.

Alternativamente, ha surgido programas multicliente que soportan varios protocolos como Pidgin o Trillian. Recientemente, algunos servicios de mensajería han comenzado a ofrecer telefonía IP (VoIP), videoconferencia, que permiten integrar capacidades de transmitir audio y vídeo junto con las palabras.

La mensajería instantánea requiere el uso de un cliente de mensajería instantánea que realiza el servicio. La mayoría de los servicios presentan el "aviso de presencia", indicando cuando el cliente de una persona en la lista de contactos se conecta o en qué estado se encuentra, si está disponible para tener una conversación. Permite mostrar mensaje de estado y enviar archivos. Los clientes de mensajería instantánea más utilizados son ICQ, Yahoo! Messenger, Windows Live Messenger, Gtalk y AIM.

1.3.1 Moodle y la Mensajería Instantánea

Las herramientas de comunicación son fundamentales en cualquier plataforma educativa ya que facilitan y enriquecen la interacción entre los usuarios. Dentro de Moodle la mensajería instantánea es importante en el proceso de enseñanza que se lleva a cabo a través de la plataforma, pues permite la comunicación síncrona entre los estudiantes y los tutores permitiendo la resolución de problemas en un tiempo real, esclarecer dudas que se presenten, creando así un ambiente colaborativo.

Sin embargo, esta actividad presenta deficiencias en su funcionamiento, lo cual puede impedir que se realice de manera eficiente el proceso de enseñanza-aprendizaje en la plataforma. El alto consumo de recursos que produce la utilización del chat reduce la calidad y eficiencia de las comunicaciones así como el funcionamiento de la plataforma. Otro de los problemas que presenta son los módulos que se integran a clientes de jabber externos, los cuales en su mayoría requieren de acceso a internet.

1.3.1.1 Módulos de Moodle para Mensajería Instantánea

La plataforma Moodle para mantener una comunicación síncrona hace uso de los módulos de mensajería instantánea. Los módulos son componentes auto-controlados que incorporan nuevas funcionalidades a una aplicación. Estos deben ser fáciles de instalar y mantener, asegurando aislar el impacto que pueda tener un fallo sobre el resto del programa. La lógica del funcionamiento de Moodle se encierra en dos grandes grupos de módulos: recursos y actividades. (de la Torre, 2006)

Módulos de recursos: Representan los contenidos y materiales del curso. Son todo tipo de textos, libros, apuntes, presentaciones de diapositivas, enlaces a páginas web externas etc., pensados para que los estudiantes los lean y estudien sobre ellos.

Módulos de actividades: Son la parte activa y colaborativa donde el estudiante tiene que hacer algo más que leer un texto. Debates y discusiones, resolución de problemas propuestos, redacción de trabajos, talleres, cuestionarios en línea, etc.

A continuación se explica brevemente el módulo chat.

Este módulo permite una interacción fluida mediante texto síncrono. Incluye las fotos de los perfiles en la ventana de chat. Soporta direcciones URL, emoticonos, integración de HTML e imágenes. Todas las sesiones quedan registradas para verlas posteriormente, y pueden ponerse a disposición de los estudiantes.

El chat de la plataforma Moodle presenta dificultades como son el alto consumo de recursos que causa esta actividad, el cual reduce considerablemente el rendimiento de la misma afectando a muchos de los usuarios ya que no reciben los mensajes que les son enviados. Además, en ocasiones los usuarios al intentar conectarse no pueden acceder a esta actividad y en las salas de conferencias los mensajes enviados no son mostrados o no deja escribir ningún texto. Otro de los problemas presentes es que su interfaz cuenta con un nivel de desarrollo muy precario con respecto al desarrollo actual de los chats.

Moodle en su desarrollo ha tenido distintos módulos y bloques de chat aportados por la comunidad, entre los que se pueden mencionar:

Small Hack

Este es un hack para el chat de Moodle que no requiere de tablas de bases de datos adicionales. Es impulsado por Ajax, utiliza div en vez de iframes y los mensajes y los usuarios se actualizan automáticamente. Tiene algunos sonidos y otras opciones. Pero presenta una limitación la cual es expresada por el autor “*no puede ser producido, es solo un experimento*” (St, 2009).

Capítulo 1: Fundamentación Teórica

Este chat presenta poca documentación, además de numerosos errores planteados por los usuarios los cuales no han sido resueltos. La última actualización del código fue realizada hace un año y no se le habían solucionado las deficiencias presentes. Por todo esto la aplicación no es muy aconsejable a la hora de utilizar Moodle (Moodle, 2010).

123 Flash Chat

123 Flash Chat Server implementa servidores de mensajería instantánea en páginas web. El servidor se ha desarrollado con lenguaje Java, el cliente en cambio funciona en flash y su interfaz tiene amplias posibilidades de personalización.

La tecnología de este servidor permite organizar sesiones de trabajo, reuniones en línea, videoconferencias, etc. además de soportar la actividad de miles de usuarios que por otra parte, se controlan gracias a la compatibilidad con bases de datos como PHP-Nuke, Post-Nuke, Asp-Nuke, etc.

Características (123flashchat, 2010)

- Un alto rendimiento: alta fiabilidad y soporte de alta carga.
- Integración completa con base de datos externa.
- Respuesta rápida y fácil de usar.
- Penetración en los cortafuegos.
- Comunicación segura con soporte multilingüe y autodetección del idioma.
- Capacidad para registrar el historial de conversación y publicar las transcripciones del chat.
- Contraseña para proteger la sala de chat y nombrar un sólo miembro por sala.
- Capacidad para especificar el número máximo de usuarios por cada habitación y el "tiempo máximo de inactividad" para los usuarios.

Skype

Es un programa de software creado por los empresarios suecos y daneses Niklas Zennström y Janus Friis. Skype permite a los usuarios hacer llamadas telefónicas a través de Internet a otros usuarios de Skype de forma gratuita, o a teléfonos fijos y teléfonos móviles por una tarifa. Las características adicionales incluyen mensajería instantánea, transferencia de archivos, servicio de mensajes cortos, videoconferencias y su capacidad para eludir cortafuegos.

En Moodle se desarrolló un módulo donde se integraba Skype con la plataforma y así beneficiarse de las características de esta herramienta. Se pueden hacer vídeo llamadas, enviar archivos, chatear y enviar sms.

Capítulo 1: Fundamentación Teórica

Skype posee ventajas como la calidad en el sonido, es fácil de usar, gratuito, sin publicidad, sin spyware. Además, permite contactar con una persona que tiene conexión a internet, te puedes comunicar gratuitamente con ella. Este presenta deficiencias como el mal manejo de la red por los usuarios, llamada la voz se tarda en llegar y así mismo en retornar, se recomienda banda ancha para su uso y no se ejecuta en equipos Win9x (SKYPE, 2011).

Chat Users

Este bloque funciona de la misma manera que el bloque de usuarios en línea, excepto que muestra una lista de los usuarios en una determinada actividad de chat. El bloque es configurable, ya que puede cambiar el título y seleccionar la actividad de chat para buscar usuarios.

Características

- **Compatible:** funciona en Internet Explorer (versión 6 o superior). Mozilla Firefox, Opera, Safari, Google Chrome.
- **Escalable:** no utiliza los recursos del servidor. El servidor proporciona la velocidad, el tiempo de actividad y las actualizaciones.
- **Idioma personalizable:** Permite editar el lenguaje de toda la interfaz del chat.
- **Ventanas múltiples:** Permite tener abiertas varias ventanas del chat dentro del sitio web.
- **Auto-enlaces:** Urls en los mensajes se convierten automáticamente en enlaces.

1.3.2 La Mensajería Instantánea en otros LMS

En la actualidad existen numerosas plataformas educativas (libres y privadas) que apoyan el proceso de enseñanza aprendizaje. Todos estos LMS presentan módulos que permiten la comunicación síncrona entre los usuarios, pero gran cantidad de estos son tecnologías dependientes de la arquitectura que presenta la plataforma en su mayoría programadas con PHP y accesos recurrentes a bases de datos que disminuyen considerablemente el rendimiento.

Por otra parte, muchos de estos LMS presentan módulos que se integran a clientes de chat en internet elaborados exclusivamente para estos sistemas los cuales mejoran los problemas de rendimiento pero en el caso de nuestro país no es factible debido al poco acceso que tenemos al mismo. También están las plataformas que utilizan la tecnología AJAX la cual mejora la calidad de la comunicación de estos clientes aunque siguen siendo dependientes de la arquitectura presente en la plataforma. Además,

Capítulo 1: Fundamentación Teórica

presentan poca información referente a como utilizaron la tecnología siendo de poca utilidad.

1.4 Tecnologías para la implementación de sistemas de mensajería instantánea

En la actualidad, existen diversas tecnologías para la implementación de sistemas de mensajería instantánea. Son bastas las aplicaciones y librerías que haciendo uso de protocolos de comunicación permiten desarrollar chat de manera fácil y eficiente, e incorpora características que los dotan de gran facilidad de uso, un entorno agradable y de una serie de atributos que le otorgan robustez y seguridad a estos sistemas.

1.4.1 Protocolos para Mensajería Instantánea

Para hacer posible el intercambio fiable y eficaz de información entre ordenadores y otros dispositivos, es necesario disponer de un conjunto de reglas y normas denominado protocolo. Sin protocolos que controlen el intercambio de datos entre 2 puntos de la red sería imposible establecer y mantener una comunicación. Un protocolo define el formato en el que se van a enviar los datos, y se encarga de controlar el tráfico en la red. Una transmisión, por lo tanto, sólo puede efectuarse cuando los dispositivos utilizan el mismo protocolo.

Los protocolos no funcionan de forma aislada, normalmente, cooperan varios protocolos para llevar a cabo la comunicación. Por ejemplo un protocolo será responsable de controlar el flujo de tráfico en cada canal, y otro protocolo puede seleccionar el canal entre varios candidatos para que lo utilice el primer protocolo en el caso de las LAN o seleccionar el mejor canal entre diferentes alternativas en el caso de las WAN para ser utilizado por el primer protocolo (Cancelo López, y otros, 2007).

Existen diferentes tipos de protocolos que son usados en la actualidad para la implementación de sistemas de mensajería instantánea entre los que se encuentran:

HyperText Transfer Protocol

Protocolo que enlaza a través de hipervínculos, las páginas de Hipertexto (HTML), que forman el World Wide Web. El protocolo HTTP consiste en un conjunto de reglas que se aplican a las peticiones que hace un cliente o navegador y a las respuestas que entrega un servidor (Rendón H, 2007).

HTTP es un protocolo de tipo solicitud-respuesta incluido dentro de la familia de los protocolos TCP/IP que se utiliza en Internet. Esto quiere decir que, cada vez que accedemos aun recurso a través de este, se establece una conexión diferente e independiente de las anteriores. Cuando se teclea una dirección o se pincha un

Capítulo 1: Fundamentación Teórica

enlace, el navegador establece una conexión TCP con el servidor web al que pertenece la dirección. Una vez establecida la conexión el cliente envía un mensaje al servidor (la solicitud) y este responde con otro mensaje (L respuesta) (Berzal, y otros, 2005).

Las principales características del protocolo HTTP (Romero Laguillo, 1998):

- *“Toda la comunicación entre clientes y servidores se realiza en modo binario con caracteres de 8 bits. De esta forma, se puede transmitir cualquier tipo de documento, ya sea texto o binario, respetando su formato original*
- *Permite la transferencia de objetos multimedia. El contenido de cada objeto intercambiado está identificado por su clasificación MIME.*
- *Los tres verbos básicos más utilizados por un cliente para dialogar con un servidor: GET, para recoger un objeto, POST, para enviar información al servidor y HEAD, para solicitar las características de un objeto.”*

Este protocolo puede ser utilizado para la creación de sistemas de mensajería instantánea, pero su principal problema es que sobrecarga el sistema debido a que es necesario realizar constantemente consultas al servidor en busca de nuevos mensajes publicados, produciendo un alto consumo de recursos.

Oscar

Oscar es el protocolo oficial del programa de mensajería instantánea de AOL, AIM. Por ser un desarrollo propietario, muchos diseñadores de programa con soporte para múltiples plataformas de mensajería han tenido que recurrir a la ingeniería inversa para conocer su forma de actuar y adaptar sus programas para hacerlos compatibles con AIM.

Oscar funciona en tres pasos, primero realiza la autenticación del usuario, después entra en funcionamiento el sistema de envío y recepción de datos (BOS) y finalmente el navegador del Chat (ChatNav) para crear la sala de charlas. Este protocolo no se utilizará en el desarrollo del trabajo debido a que no ofrece ninguna documentación ni da acceso a sus fuentes.

Internet Relay Chat

El protocolo IRC (Internet Relay Chat) permite realizar conversaciones en tiempo real a través de Internet, aunque normalmente se utiliza para conversar en línea, también sirve para transmitir ficheros. El IRC se basa en la existencia de servidores, adonde se conectan los usuarios para mantener conversaciones. Los participantes necesitan un

Capítulo 1: Fundamentación Teórica

programa cliente para conectarse a los servidores, que pueden ser MIRC, Microsoft Chat y WSIRC, entre otros. A su vez, cada servidor de IRC está vinculado a otros servidores formando redes. Las últimas versiones de programas cliente de este sistema permiten la transmisión de ficheros y sonido.

El IRC organiza sus conversaciones en canales. Después de conectarse al servidor, se mira la lista de canales disponibles y se elige aquel al que quiere unirse. En cualquier momento, podemos añadirnos a la conversación o bien abandonarla, si esta no nos interesa.

Limitaciones del Protocolo IRC (España Boquera, 2003):

- *“ Problemas de escalado*
El modelo distribuido obliga a cada servidor a disponer de información de estado de carácter global, siendo esta la limitación más seria del protocolo, ya que se acota la extensión máxima que una conversación puede alcanzar. El requisito de que todos los servidores conozcan el estado de los restantes, a todos los clientes y a todos los canales implica un gran intercambio de información, ya que cualquier novedad que se produzca en el sistema debe actualizarse inmediatamente. Este problema resulta más acuciante a medida que crece el número de usuarios que se conectan a la red IRC.
- *Fiabilidad*
Tal y como se ha dispuesto la topología de la red, existe un único camino entre dos servidores. Como consecuencia de ello, si uno de los enlaces que los une falla, se interrumpe irremisiblemente la comunicación”.

Otro punto desfavorable de este protocolo es su vulnerabilidad frente a situaciones de congestión, puesto que no se dispone de rutas alternativas para aliviarla.

Extensible Messaging and Presence Protocol

El protocolo XMPP (Mensajería Extensible y Protocolo de Presencia), es un protocolo abierto y extensible para la comunicación en tiempo real, que alimenta una amplia gama de aplicaciones, incluyendo mensajería instantánea, voz y vídeo y enrutamiento de datos XML. Además, posee muchas implementaciones abiertas de servidores, clientes y librerías para las más diversas plataformas y lenguajes. Este protocolo en su funcionamiento topológico se basa en la clásica arquitectura cliente – servidor, aunque no fuerza a hacerlo así y mediante TLS permite cifrar los mensajes empleando diferentes algoritmos como RSA y DSS (XMPP, 2010).

Capítulo 1: Fundamentación Teórica

Características

Entre las características más significativas que posee el protocolo en cuanto a adaptabilidad y sencillez se tienen las siguientes (XMPP, 2010):

- **Abierto, público y libre**
El protocolo XMPP está ampliamente documentado y puesto a disposición del público por la Jabber Foundation, en el sitio web de la misma. Además, cualquier individuo, organización o empresa pueden hacer uso de él para sus tecnologías de comunicación sin tener que pagar derechos de autor o licenciarlo gracias a que es abierto y multiplataforma.
- **Estándar**
XMPP está avalado por la organización imparcial IETF que se encarga de su registro y conservación. Al ser un estándar que ha seguido un riguroso proceso de regularización, no es concebible que cambie de la noche a la mañana sin previo aviso, por ejemplo, por los intereses comerciales de una empresa.
- **Probado**
Numerosas organizaciones privadas, grupos de usuarios y universidades usan la tecnología XMPP como sistema de comunicación, siendo notable la Universidad Jaume I de Castellón, que alberga uno de los servidores Jabber más grandes de España.
- **Descentralizado**
Su topología es distribuida. Cualquier individuo u organización pueden fácilmente montar un servidor para dar servicio a una comunidad de usuarios en régimen público o privado.
- **Seguro**
Las comunicaciones punto a punto, pueden ir aseguradas bajo una capa segura SSL. De este modo, se cifra la información que se intercambia. Adicionalmente, es habitual el uso de una firma o cifrado GPG (o PGP) de cada mensaje para autenticar la autoría del mismo por parte de los clientes.
- **Extensible**
XMPP se apoya en una potente característica de XML, los espacios de nombres. Al comportarse un servidor de XMPP básicamente como un router de XML, basta con definir nuevos espacios de nombres sobre los ya existentes para extender las funcionalidades de Jabber sin modificar en ningún momento

Capítulo 1: Fundamentación Teórica

los servidores existentes. El "peso" del protocolo recae en los clientes, que son los que interpretan los datos XML que reciben.

- Flexible
XMPP es un potente y versátil protocolo que permite a dos aplicaciones (cualesquiera) comunicarse eficientemente usando XML

En el desarrollo de este trabajo se empleará el protocolo XMPP por su adaptabilidad, su robusto sistema de seguridad, además de ser multiplataforma y un estándar libre. Además, este protocolo es uno de los más utilizados para implementar sistemas de mensajería instantánea por lo cual presenta abundante documentación.

1.4.2 Sistemas y librerías para la implementación de servicios de mensajería instantánea

HOLLA

Holla es un completo y moderno sistema de chat gratuito y de código abierto realizado con JavaScript y AJAX. Esta aplicación nos permite crear diferentes salas de chat, eliminarlas, invitar a otros participantes, compartir archivos y mucho más.

Todas las funciones se controlan mediante una agradable y moderna interfaz, la cual es sumamente fácil de usar: por ejemplo, para compartir un archivo, sólo tendremos que arrastrarlo y soltarlo en la ventana de conversación. Todas las personas conectadas a esa sala de chat pueden ver la barra de progreso y acceder al archivo una vez finalizada la carga.

Holla es una herramienta que puede utilizarse por ejemplo para la comunicación entre grupos de trabajo, aunque obviamente no tiene límites en cuanto a su uso (Adictosalared, 2010).

PYTHON-IRCLIB

Esta biblioteca hecha para python se destina a encapsular el protocolo IRC en un nivel bastante bajo. Proporciona un marco de cliente IRC por eventos. Cuenta con un apoyo bastante completo para el protocolo básico de IRC, CTCP y conexiones DCC (PYTHON, 2005).

La biblioteca simplifica en gran medida la creación de aplicaciones que interactúan a través de IRC mediante la abstracción del protocolo. Incluso contiene código que se ocupa específicamente de la creación de robots que puedan comunicarse y recibir órdenes de los miembros del servidor IRC determinado.

Capítulo 1: Fundamentación Teórica

Características (PYTHON, 2005):

- La abstracción del protocolo IRC.
- Maneja múltiples conexiones de servidor de IRC.
- Los mensajes para el servidor de IRC se realizan mediante llamadas a métodos en un objeto de conexión de IRC.
- Los mensajes de un servidor de IRC desencadena los acontecimientos, que pueden ser capturados por los controladores de eventos.
- Las funciones pueden estar registradas para ejecutarse en momentos determinados por el evento de circuito.

Limitaciones actuales:

- Los datos no se escriben de forma asíncrona con el servidor, es decir, la escritura puede bloquearse si los buffers TCP se rellenan.

XMPPHP

XMPPHP es una librería escrita en PHP, es software libre, puede redistribuirlo y / o modificar bajo los términos de la GNU General Public License publicada por la Free Software Foundation, bien de la versión 2 de la licencia, o cualquier versión posterior.

Permite realizar acciones sobre el protocolo XMPP y realizar tareas cómo enviar mensajes y alertas a los usuarios por medio de Google Talk o Jabber.

Algunas de las características incluyen (HostingGoogleProject, 2011):

- Permite conectarse a cualquier servidor XMPP 1.0 (Google Talk, Talk LJ, jabber.org, etc.)
- Soporta el cifrado TLS.
- Varios métodos de procesamiento XML y estilos admitidos (proceso indefinidamente, processUntil un evento, processTime durante varios segundos), a la espera de acontecimientos o de mapa de ellos, etc.

JSJAC

Es una librería escrita en JavaScript para facilitar la implementación de clientes de jabber basados en web. Para la comunicación con los servidores de jabber necesita del HTTP Polling o del HTTP Binding y se realiza usando XML HTTP Request object conocido como la tecnología AJAX. JSJAC constituye la base en la que están implementados múltiples clientes jabber/XMPP sobre tecnologías web siendo una de las librerías más utilizadas con este fin.

Compatible con varios navegadores tales como:

Capítulo 1: Fundamentación Teórica

- Internet Explorer.
- Mozilla Firefox.
- Opera.
- Safari

JSJAC presenta un diseño orientado a objeto que permite que el trabajo con esta librería se realice de manera sencilla. Además, se comunica con el servidor de jabber a través del protocolo XMPP haciendo que las comunicaciones sean seguras y eficientes. Las clases que se definen en esta librería son fáciles de utilizar en la implementación, pues cada una se relaciona con un caso específico que se utilizan en la creación de un cliente de jabber.

En este trabajo se utilizará la librería JSJAC por su fácil uso, además de que utiliza el protocolo XMPP para su comunicación. Es multiplataforma y compatible con varios navegadores. También presenta abundante documentación en la cual se puede evacuar cualquier duda o ver ejemplos relacionados con su utilización en la implementación de clientes de jabber.

IJAB

IJAB es un sistema que permite añadir chat en aplicaciones web, con la característica que usa XMPP/Jabber para realizar las comunicaciones. Es software libre y los usuarios pueden gestionar su propia mensajería instantánea, sin limitaciones o costes de software, complementando la filosofía XMPP de una red distribuida. IJAB es interpretado exclusivamente por el navegador web del cliente. Se ejecuta únicamente en un navegador y no requiere la descarga o instalación de software adicional.

Capítulo 1: Fundamentación Teórica

Entre las características encontramos (IJAB, 2010):

- Basado en el protocolo XMPP: lo cual permite usar servidores basados en ese protocolo.
- Soporte para MSN y AIM, añadiendo transportes al servidor XMPP.
- Totalmente JavaScript.
- Compatible con Firefox, IE, Chrome y Safari.
- Modos de aplicación: stand-alone, web chat y live.
- Sin pop-ups.
- Notificaciones de nuevos mensajes.
- Sonido.
- Múltiples usuarios en el chat (MUC).
- Búsqueda de usuarios.
- Administración: permite añadir y borrar usuarios, y administrar grupos.

Además de estar hecha en Java script utiliza Google Gears, soporta temas y es compatible con Live Messenger y permite la creación de grupos de chat cerrados para nuestra propia comunidad. Desde luego es una opción a tener en cuenta si necesita integrar en tu página web o proyecto un servicio de mensajería.

1.5 Lenguajes de Programación para implementar sistemas de mensajería instantánea.

PHP

PHP, acrónimo de Hypertext Preprocessor. Es un lenguaje interpretado, diseñado originalmente para la creación de páginas web dinámicas, y con acceso a información almacenada en bases de datos.

La naturaleza de código abierto (open-source) de PHP, lo muestra como una alternativa de fácil acceso para todos. Una de las ventajas más notables resulta el consumir pocos recursos, por lo que generalmente se ejecuta con rapidez. Es capaz de conectarse con la mayoría de los manejadores de bases de datos que se utilizan en la actualidad.

PHP contiene funciones de seguridad que previenen la inserción de órdenes dentro de una solicitud de datos, posee manejo de excepciones, posibilita la generación de archivos PDF, flash, así como imágenes en diferentes formatos. Además es el lenguaje en el que está desarrollada la plataforma Moodle siendo de vital importancia su utilización en el desarrollo del presente trabajo.

Capítulo 1: Fundamentación Teórica

JavaScript

JavaScript es un lenguaje de programación del lado del cliente ampliamente utilizado en el mundo del desarrollo web por ser muy versátil y compatible con la mayoría de los navegadores modernos, además, este añade al HTML movimiento e interactividad con el usuario.

Con Java Script se pueden definir interactividades con el usuario y crear efectos en las páginas web. El navegador es el encargado de interpretar las instrucciones Java Script y ejecutarlas, de modo que el recurso más costoso con que cuenta este lenguaje es un navegador compatible con JavaScript (Van Lancker, 2005).

JavaScript tiene algunas limitaciones, las cuales derivan de su objetivo principal, que es adicionar interactividad a las páginas web (Figueroa, 2011):

- *“Trabaja enteramente dentro del navegador del cliente y no se comunica con el servidor ni con otras máquinas.*
- *No puede crear sus propios gráficos, solamente puede manipular gráficos hechos en otros programas.*
- *Trabaja de forma diferente en diferentes navegadores.”*

JavaScript presenta la tecnología Ajax (Asynchronous JavaScript y XML) que es un concepto de programación basados en múltiples tecnologías Web como JavaScript y XML - de ahí el nombre de AJAX. La idea de AJAX es conectar a una página web con un servidor Web sin que recargue la página. Es por eso que se utiliza AJAX, porque será el responsable de establecer la conexión entre la página web y el servidor.

AJAX propone el desarrollo de aplicaciones Web de la siguiente forma (Pérez, 2008):

- *“Cambios dinámicos en la visualización del contenido mostrado y control de eventos del usuario a través del DOM.*
- *Intercambio de datos y manipulación usando XML y XSLT.*
- *Obtención de datos de forma asíncrona a través de XMLHttpRequest*
- *Presentación basada en estándares: XHTML y CSS.”*

“JavaScript también utiliza CCS, el cual es un lenguaje de hojas de estilos creado para controlar el aspecto o presentación de los documentos electrónicos definidos con HTML y XHTML. CSS es la mejor forma de separar los contenidos y su presentación y es imprescindible para crear páginas web complejas” (Pérez, 2008).

Además, facilita la portabilidad de los contenidos porque una vez que el contenido HTML es aplicado correctamente, es fácil cambiar el aspecto de su sitio.

Capítulo 1: Fundamentación Teórica

JavaScript es uno de los lenguajes más utilizados en el desarrollo de clientes jabber, además, cuenta con la tecnología Ajax la cual permite que las aplicaciones tengan un mayor rendimiento. También presenta una amplia documentación con la cual se pueden evacuar las dudas que surjan durante el desarrollo del trabajo.

HTML

HTML (HyperText Markup Language) es el lenguaje de marcado de hipertexto predominante para la construcción de páginas web. Es usado para describir la estructura y el contenido en forma de texto, así como para complementar el texto con objetos tales como imágenes. HTML es un lenguaje de composición de documentos y especificación de unión de hipertexto que define la sintaxis y coloca instrucciones especiales que no muestra el navegador, aunque sí le indica cómo desplegar el contenido del documento, incluyendo texto, imágenes y otros medios soportados.

Aunque HTML es un estándar muy usado se considera un lenguaje estático basado en etiquetas muy limitado que no puede ser ampliado. Efectos como drag & drop (arrastrar y soltar) y cambio de tamaño de elementos son imposibles de realizar, al igual que la actualización de zonas específicas de la página sin la necesidad de tener que redireccionar una nueva página HTML (Musciano, y otros, 2006).

XHTML

XHTML (Extensible Hypertext Markup Language) es una versión más estricta y limpia de HTML, que nace precisamente con el objetivo de remplazar a HTML ante su limitación de uso con las cada vez más abundantes herramientas basadas en XML.

Las ventajas del XHTML son (Van Lancker, 2007):

- La vida de los documentos. Las paginas que elabore de acuerdo con los últimos estándares de la Web, tendrán un código válido durante muchos años.
- Una compatibilidad mayor con los exploradores. Una norma común, es siempre mejor que un código que solo funciona en uno o dos exploradores.
- Una mejor legibilidad del código, sobre todo desde el punto de vista de las actualizaciones de la página.
- Una portabilidad mayor de las páginas hacia otros soportes (PDA, portátiles y otros aparatos de bolsillo). Un documento XHTML correctamente estructurado puede readaptar su formato con facilidad para poder visualizarse en diferentes aparatos.

Capítulo 1: Fundamentación Teórica

PYTHON

Python es un lenguaje interpretado e interactivo. Incorpora módulos, excepciones, tipos dinámicos, tipos de muy alto nivel dinámico de datos, y las clases. Este lenguaje orientado a objetos combina potencia con una sintaxis muy clara. Tiene interfaces a muchas llamadas al sistema y bibliotecas, así como a diversos sistemas de ventana, y es extensible en C o C + +. También es utilizable como un lenguaje de extensión para aplicaciones que necesitan una interfaz programable.

Características (PYTHON, 2005):

- **Fácil de aprender:** Python es fácil porque tiene una sintaxis extraordinariamente simple.
- **Libre y de Código Abierto:** Python es un ejemplo de un FLOSS (Free / Libre and Open Source Software). Se puede distribuir libremente copias de este software, leer su código fuente, realizar cambios en él y utilizar sus piezas de uso en nuevos programas libres.
- **Portátil:** Debido a su naturaleza de código abierto, Python ha sido portado (es decir cambiar para que funcione) a muchas plataformas. Todos los programas Python pueden trabajar en cualquiera de estas plataformas sin requerir ningún cambio en absoluto. Se puede utilizar Python en Linux, Windows, FreeBSD, Macintosh, Solaris, OS / 2, Amiga, AROS, AS/400, BeOS, OS/390, z / OS, Palm OS, QNX, VMS, Psion, Acorn RISC OS, VxWorks, PlayStation, Sharp Zaurus, Windows CE, e incluso PocketPC!
- **Orientada a Objetos:** Python admite la programación orientada a objetos porque el programa se construye alrededor de los objetos que se combinan los datos y la funcionalidad. Python tiene una forma muy poderosa pero simple de hacer programación orientada a objetos, especialmente cuando se compara con grandes lenguas como C + + o Java.
- **Integrable:** Se puede integrar Python dentro de la carpeta C / C + + para dar 'scripting' capacidades para los usuarios de su programa.

JAVA

Java es un lenguaje que tiene semejanzas con C y C++. Java es un lenguaje orientado a objetos, dichos objetos agrupan en estructuras encapsuladas tanto sus datos como los métodos (o funciones) que manipulan esos datos. Java proporciona una colección de clases para su uso en aplicaciones de red, que permiten abrir sockets y establecer y aceptar conexiones con servidores o clientes remotos, facilitando así la creación de aplicaciones distribuidas. Java soporta sincronización de múltiples hilos de ejecución

Capítulo 1: Fundamentación Teórica

(multithreading) a nivel de lenguaje, especialmente útiles en la creación de aplicaciones de red distribuidas.

El lenguaje Java es muy dinámico en la fase de enlazado, sus clases solamente actuarán en medida en que sean requeridas o necesitadas; con esto permitirá que los enlaces se puedan incluir incluso desde fuentes muy variadas o desde la red. Java es compatible con los más variados entornos de red, cualquiera sean estos desde Windows 95, Unix a Windows NT y Mac. Es muy versátil ya que utiliza byte-codes que es un formato intermedio que sirve para transportar el código eficientemente. Por ser indiferente a la arquitectura sobre la cual está trabajando, esto hace que su portabilidad sea muy eficiente, sus programas son iguales en cualquiera de las plataformas, ya que Java especifica tamaños básicos, esto se conoce como la máquina virtual de java.

1.6 Herramientas para implementar sistemas de mensajería instantánea basados en web.

Zend Studio

Zend Studio es Entorno de Desarrollo Integrado (IDE) disponible para desarrolladores profesionales, proporcionando las capacidades que se necesitan para desarrollar aplicaciones de negocios. Características como: reestructuración de código fuente, generación de código, asistencia de código y análisis semántico, se combinan para permitir un rápido desarrollo de aplicaciones. Esta herramienta proporciona una completa depuración de programación y soporte de pruebas que lo ayuda a analizar y resolver rápidamente los problemas. Una integración justa con Zend Server acelera el análisis de la causa raíz de los problemas detectados en la fase de prueba, montaje o entorno de producción.

Zend Studio mejora la productividad con características de desarrollo orientada a equipos, tales como soporte para el manejo de configuración fuente (CVS, SVN) y montajes de proyectos compartidos. Estas características ayudan a mejorar la colaboración entre miembros de proyecto. Es el IDE PHP más actualizado que da soporte a las tecnologías más recientes tales como PHP 5.3, Zend Framework 1.10, el último Eclipse Platform (Helios), asegurando que su entorno esté siempre actualizado y adelantado. Esta herramienta a pesar de ser privativa se ha realizado versiones para Windows, Linux y MacOS aunque el desarrollo de las versiones de este último sistema se retrase un poco más.

NetBeans

NetBeans es un entorno de desarrollo visual de código abierto que permite programar aplicaciones principalmente en Java además de lenguajes como Ruby, C/C++ o PHP.

Capítulo 1: Fundamentación Teórica

Otro aspecto importante de Netbeans es que funciona en diversos sistemas operativos como Windows, Mac, Linux o Solaris.

Con la utilización del NetBeans no solo es posible elaborar potentes aplicaciones para el Escritorio, también para la Web y para dispositivos portátiles, como móviles o Pocket PC, sin que cambie la forma de programar. La programación mediante este IDE se realiza a través de componentes de software modulares, también llamados módulos los cuales aportan muchas funcionalidades.

Como principal ventaja de NetBeans se tiene que a diferencia de otras herramientas como el Zend Studio no es un software propietario que requiere una licencia de pago, presenta un entorno de desarrollo visual de código abierto. NetBeans además, facilita la programación a través de componentes de software modulares y posibilita integración con PHP y la librería jQuery de JavaScript. Por todo lo anteriormente mencionado se elige NetBeans para el desarrollo de la solución que se propone.

1.7 Proceso de desarrollo de software

El proceso de desarrollo de software "es aquel en que las necesidades del usuario son traducidas en requerimientos de software, estos requerimientos transformados en diseño y el diseño implementado en código, el código es probado, documentado y certificado para su uso operativo". Concretamente "define quién está haciendo qué, cuándo hacerlo y cómo alcanzar un cierto objetivo". (Jacobson, y otros, 2004)

Un proceso de desarrollo de software tiene como propósito la producción eficaz y eficiente de un producto software que reúna los requisitos del cliente.

No existe un proceso de software único, universal que sea efectivo para todos los contextos de proyectos de desarrollo. Debido a esta diversidad, es difícil automatizar todo un proceso de desarrollo de software por lo que se hace necesario contar con una metodología que garantice cumplir con los planes de producción del software y la satisfacción del cliente para que se puedan alcanzar los objetivos del proyecto.

1.7.1 Metodología de Desarrollo de Software

Un proceso de software detallado y completo suele denominarse "Metodología", define Quién debe hacer Qué, Cuándo y Cómo debe hacerlo. Las metodologías se desarrollan con el objetivo de dar solución a los problemas existentes en la producción de software, que cada vez son más complejos. Estas engloban procedimientos, técnicas, documentación y herramientas que se utilizan en la creación de un producto de software. No existe una metodología de software universal. Las características de cada proyecto (equipo de desarrollo, recursos, etc.) exigen que el proceso sea configurable.

Capítulo 1: Fundamentación Teórica

La comparación y/o clasificación de metodologías no es una tarea sencilla debido a la diversidad de propuestas y diferencias en el grado de detalle, información disponible y alcance de cada una de ellas.

A grandes rasgos, si se toma como criterio su filosofía de desarrollo, aquellas metodologías con mayor énfasis en la planificación y control del proyecto, en especificación precisa de requisitos y modelado, reciben el apelativo de Metodologías Tradicionales (o peyorativamente denominada Metodologías Pesadas, o Peso Pesado).

Otras metodologías, denominadas Metodologías Ágiles, están más orientadas a la generación de código con ciclos muy cortos de desarrollo, se dirigen a equipos de desarrollo pequeños, hacen especial hincapié en aspectos humanos asociados al trabajo en equipo e involucran activamente al cliente en el proceso.

Entre las metodologías ágiles y tradicionales más reconocidas se encuentran la Programación Extrema (XP) y el Proceso Unificado de Rational (RUP) respectivamente.

1.7.1.1 Programación Extrema

La programación extrema, o XP, es una metodología ágil concebida para dirigir las necesidades específicas del desarrollo de software conducido por equipos pequeños. Se define como especialmente adecuada para proyectos con requisitos imprecisos y muy cambiantes, y donde existe un alto riesgo técnico. Esta metodología está centrada en potenciar la comunicación entre el desarrollador y el cliente desde el primer día.

Es considerada ligera, flexible, predecible, de bajo riesgo, y no por ello menos científica. Entre otras ventajas pueden mencionarse los pocos requerimientos de documentación y planificación, así como la exigencia de tener siempre el cliente disponible para el desarrollo, implicando una mejor correspondencia entre el producto y la necesidad del negocio.

1.7.1.2 Proceso Unificado de Desarrollo

RUP es un proceso bien definido, estructurado y adaptable a las características y necesidades de cada proyecto específico. Utiliza el Lenguaje Unificado de Modelado (UML) para preparar todos los esquemas de un sistema software. Este proceso unificado tiene tres características distintivas:

- **Dirigido por Casos de Uso:** Los casos de uso reflejan lo que los usuarios futuros necesitan y desean, lo cual se capta cuando se modela el negocio y se

Capítulo 1: Fundamentación Teórica

representa a través de los requerimientos. A partir de aquí los casos de uso guían el proceso de desarrollo.

- **Centrado en Arquitectura:** La arquitectura muestra la visión común del sistema completo en la que el equipo de proyecto y los usuarios deben estar de acuerdo, por lo que describe los elementos del modelo que son más importantes para su construcción, los cimientos del sistema que son necesarios como base para comprenderlo, desarrollarlo y producirlo económicamente.
- **Iterativo e Incremental:** RUP propone que cada fase se desarrolle en iteraciones. Una iteración involucra actividades de todos los flujos de trabajo, aunque desarrolla fundamentalmente algunos más que otros. Es práctico dividir el trabajo en partes más pequeñas o miniproyectos. Cada miniproyecto es una iteración que resulta en un incremento. Las iteraciones hacen referencia a pasos en los flujos de trabajo, y los incrementos, al crecimiento del producto. Cada iteración se realiza de forma planificada, es por eso que se dice que son miniproyectos.

RUP es uno de los procesos más generales ya que en realidad está pensado para adaptarse a cualquier proyecto. Propone 6 disciplinas ingenieriles (Modelamiento del Negocio, Requerimientos, Análisis y Diseño, Implementación, Prueba, Despliegue) y 3 de apoyo (Gestión de Configuración y Cambios, Gestión de Proyecto, Ambiente) que tienen lugar en algunas o todas las fases que define (Inicio, Elaboración, Construcción y Transición).

RUP permite trabajar con precisión y calidad, perfeccionado el software en cualquier momento del desarrollo, logrando así un sistema con la robustez necesaria, independientemente del tiempo disponible.

Si se maneja bien el conjunto de procesos es posible realizar todos los artefactos que propone RUP independientemente del tamaño del equipo de desarrollo.

Con el uso de esta metodología es posible encomendar tareas específicas en otras personas involucradas indirectamente en el proyecto; sin embargo con XP se hace necesario que los colaboradores estén involucrados completamente en el desarrollo del producto.

Por otra parte, la documentación que se obtiene mediante RUP contribuirá a lograr un mejor entendimiento del sistema por parte del equipo de desarrollo, y servirá de referencia para posteriores trabajos sobre Moodle, en cambio, XP posee pocos requerimientos de documentación y planificación.

1.7.2 Lenguaje Unificado de Modelado

“El Lenguaje Unificado de Modelado (UML) es un lenguaje de modelado visual que se usa para especificar, visualizar, construir y documentar artefactos de un sistema de software. Captura decisiones y conocimiento sobre los sistemas que se deben construir. Se usa para entender, diseñar, hojear, configurar, mantener, y controlar la información sobre tales sistemas.

UML incluye conceptos semánticos, notación, y principios generales. Tiene partes estáticas, dinámicas, de entorno y organizativas. Está pensado para ser utilizado en herramientas interactivas de modelado visual que tengan generadores de código así como generadores de informes. La especificación de UML no define un proceso estándar pero está pensado para ser útil en un proceso de desarrollo iterativo. Pretende dar apoyo a la mayoría de los procesos de desarrollo orientados a objetos.”
(Jacobson, y otros, 2004)

1.8 Herramientas CASE para el modelado

La mayoría de las herramientas CASE (Ingeniería de Software Asistida por Ordenador) hacen uso de UML. Estas herramientas se consideran aplicaciones informáticas destinadas a aumentar la productividad en el desarrollo de software de forma tal que se reduzca el coste de las mismas en términos de tiempo y dinero. Entre las más utilizadas a nivel mundial se encuentran Visual Paradigm y Rational Rose Enterprise Edition.

1.8.1 Rational Rose Enterprise Edition

Rational Rose Enterprise es el producto más completo de la familia Rational Rose, es una herramienta CASE propietaria, desarrollada por Rational Corporation, basada en el Lenguaje Unificado de Modelado. Permite crear los diagramas que se van generando durante el proceso de desarrollo del software y posee un gran número de estereotipos predefinidos que facilitan el proceso de modelación.

Rational Rose Enterprise Edition posee características adicionales como: (GSInnova, 2010)

- Soporte de ingeniería Forward y/o reversa para algunos de los conceptos más comunes de Java 1.5.
- Soporte Enterprise Java Beans™ 2.0.
- Capacidad de análisis de calidad de código.

Capítulo 1: Fundamentación Teórica

- Modelado UML para trabajar en diseños de base de datos, con capacidad de representar la integración de los datos y los requerimientos de aplicación a través de diseños lógicos y físicos.
- Integración con otras herramientas de desarrollo de Rational.
- Capacidad para integrarse con cualquier sistema de control de versiones.

1.8.2 Visual Paradigm for UML

Herramienta CASE propietaria con licencia gratuita, que soporta el ciclo de vida completo del desarrollo de software. Visual Paradigm ofrece: (Sierra, 2008)

- Diseño centrado en casos de uso y enfocado al negocio que generan un software de mayor calidad.
- Uso de un lenguaje estándar común a todo el equipo de desarrollo que facilita la comunicación.
- Capacidades de ingeniería directa (versión profesional) e inversa.
- Modelo y código que permanece sincronizado en todo el ciclo de desarrollo.
- Disponibilidad de múltiples versiones, para cada necesidad.
- Disponibilidad de integrarse en los principales IDEs.
- Disponibilidad en múltiples plataformas.

El Visual Paradigm ayuda a una más rápida construcción de aplicaciones de calidad y con menor costo. Además, permite dibujar todos los tipos de diagramas de clases, generar código desde diagramas y generar documentación.

Aunque Rational Rose Enterprise Edition es una de las herramientas más potentes para el modelado visual, es un software propietario y no es multiplataforma, inconvenientes que limitan su uso. Considerando estos aspectos, se selecciona Visual Paradigm como herramienta para el modelado de la solución que se propone, siendo esta una herramienta multiplataforma que cuenta con versiones gratuitas además, de proveer una fácil integración con el resto de las herramientas de desarrollo.

Conclusiones parciales

En este capítulo se abordaron los aspectos esenciales relacionados con el e-learning, fundamentalmente los referentes con la plataforma Moodle. Además, se realizó el análisis de las tecnologías para la implementación de sistemas de mensajería instantánea, seleccionando el protocolo XMPP y como librería se seleccionó JSJAC, la cual brinda ventajas en la implementación de clientes de jabber y utiliza el protocolo anterior para comunicarse con el servidor de jabber.

Capítulo 1: Fundamentación Teórica

También han sido expuestos los lenguajes de programación a utilizar donde se eligió PHP para la parte del servidor por ser el lenguaje en que está desarrollado Moodle y JavaScript y Ajax para la parte del cliente. Se realizó un estudio de las herramientas de desarrollo, la cual se escogió el IDE NetBeans por sus características y funcionalidades, además de ser una herramienta Open Source. Se decide utilizar como herramienta para el modelado de software el Visual Paradigm y como metodología a RUP, ya que es la metodología que se lleva a cabo en el “Centro de Desarrollo de Tecnologías para la Formación” (FORTES) del Departamento de Producción de Herramientas Educativas para el desarrollo de la plataforma Moodle.

Capítulo 2: Análisis y Diseño de un módulo para la integración de Moodle con un sistema de mensajería instantánea.

Capítulo 2: Análisis y Diseño de un módulo para la integración de Moodle con un sistema de mensajería instantánea.

Introducción

En este capítulo se describe de forma general la propuesta del sistema, la determinación de las clases que se van a emplear en su implementación, así como las relaciones entre ellas. Se especifican los requerimientos funcionales y no funcionales. Se representa el modelo de caso de uso del sistema, los diagramas de clases del análisis, y los diagramas de interacción que en este caso serán los diagramas de colaboración. Además, se realizan los diagramas del diseño web del módulo para la integración de Moodle con un sistema de mensajería instantánea sobre protocolo XMPP.

2.1 Propuesta del sistema

El ChatXMPP es un módulo para la integración de Moodle con un sistema de mensajería instantánea sobre protocolo XMPP. El cual permitirá gestionar salas de chat que posibilitarán una mejor comunicación instantánea con los actores del sistema, los cuales podrán enviar mensajes públicos. También al entrar a la sala de chat esta mostrará una lista de usuarios conectados. El sistema permitirá instalar, configurar y desinstalar el módulo.

2.2 Modelo de Dominio

Un Modelo de Dominio captura los tipos más importantes de objetos en el contexto del sistema. Los objetos del dominio representan las “cosas” que existen o los eventos que suceden en el entorno en el que funcionará el sistema. Muchos de los objetos del dominio o clases pueden obtenerse de una especificación de requisitos o mediante la entrevista con los expertos del dominio (Jacobson, y otros, 2004).

Conceptos Fundamentales

Para una mejor comprensión del modelo de dominio que se muestra en la figura 1, se realiza a continuación una explicación de las clases que lo conforman:

Moodle: El Entorno de Aprendizaje Dinámico Modular Orientado a Objetos es una plataforma para la creación de cursos y sitios web basados en Internet.

Curso: Unidad educativa en la que se ofrece un conjunto estructurado de conocimientos teóricos y/o prácticos.

Creador: Persona que posee privilegios para gestionar la Sala de Chat.

Capítulo 2: Análisis y Diseño de un módulo para la integración de Moodle con un sistema de mensajería instantánea.

Administrador: Persona que posee privilegios para gestionar el módulo.

Usuario: Persona que posee privilegios para enviar mensajes públicos.

Sala de Chat: Ventana en la cual se muestran los usuarios conectados y donde estos envían y reciben mensajes públicos o privados.

Módulo: Componente que extiende las funcionalidades de la aplicación.

A continuación se muestra el modelo de dominio de este módulo para la integración de Moodle con mensajería instantánea sobre protocolo XMPP mostrando la estructura que posee el mismo:

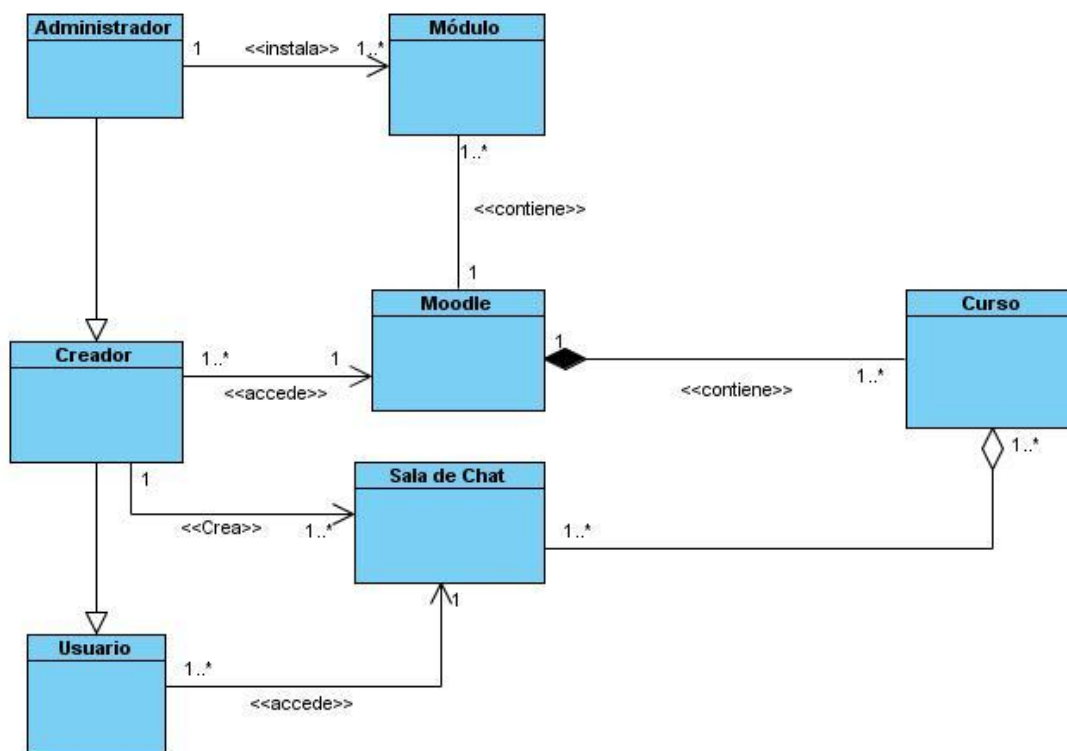


Figura 1 Modelo de Dominio

2.3 Definición de Requerimientos Funcionales

Los requerimientos funcionales de un sistema describen lo que el sistema debe hacer, es decir, describen con detalle la función del sistema, así como sus entradas y salidas, excepciones, etc (Sommerville, 2005).

Con el objetivo de lograr la creación de un módulo para la integración de Moodle con un sistema de mensajería instantánea sobre protocolo XMPP se definen a continuación los requerimientos funcionales que serán incorporados al mismo:

Capítulo 2: Análisis y Diseño de un módulo para la integración de Moodle con un sistema de mensajería instantánea.

RF 1 Instalar Módulo: El sistema debe brindar la posibilidad de instalar el módulo.

RF 2 Configurar Módulo: El sistema debe brindar la posibilidad de configurar el módulo.

RF 3 Desinstalar Módulo: El sistema debe brindar la posibilidad de desinstalar el módulo.

RF 4 Gestionar Sala de Chat: El sistema debe brindar la posibilidad de adicionar, modificar y eliminar sala de chat.

RF 4.1 Adicionar Sala de Chat: El sistema debe brindar la posibilidad de añadir una sala de chat.

RF 4.2 Modificar sala de chat: El sistema debe brindar la posibilidad de modificar la sala de chat.

RF 4.3 Eliminar sala de chat: El sistema debe brindar la posibilidad de eliminar una sala de chat.

RF 5: Autenticar Usuario: El sistema debe brindar la posibilidad de autenticarse en el servidor de mensajería instantánea.

RF 6 Entrar a la sala de chat: El sistema debe brindar la posibilidad de entrar a la sala de chat.

RF 7 Mostrar listado de usuarios conectados: El sistema debe brindar la posibilidad de mostrar el listado de usuarios conectados.

RF 8 Enviar un mensaje público: El sistema debe brindar la posibilidad de enviar un mensaje público.

2.4 Definición de Requerimientos No Funcionales

Los requerimientos no funcionales, son aquellos requerimientos que no se refieren directamente a las funciones específicas que proporciona el sistema, sino a las propiedades emergentes de éste como la fiabilidad, el tiempo de respuesta y la capacidad de almacenamiento (Sommerville, 2005).

RNF 1: Apariencia o interfaz externa: La interfaz del sistema debe ser sencilla, para que los usuarios puedan utilizar la aplicación fácilmente.

Capítulo 2: Análisis y Diseño de un módulo para la integración de Moodle con un sistema de mensajería instantánea.

RNF 2: Rendimiento: Garantizar tiempos de respuestas rápidas al igual que la velocidad de procesamiento de la información.

RNF 3: Confidencialidad: La información manejada por el sistema debe estar protegida ante el acceso no autorizado y la divulgación. El acceso al sistema se restringirá a los usuarios de acuerdo con el rol que desempeñen.

RNF 4: Disponibilidad: Se les garantizará el acceso las 24 horas del día.

RNF 5: Implementación: El sistema será implementado con la tecnología PHP y Ajax.

RNF 6: Hardware del servidor:

- Microprocesador 1.5 GHz.
- 512 MB de memoria RAM.
- 40 GB de disco duro.

RNF 7: Ayuda y documentación en línea: Orientar al usuario respecto a las opciones que se le brindan en la aplicación, a través de un sistema de ayuda que sea capaz de describir la acción que se esté realizando.

RNF 8: Portabilidad: El sistema debe ser multiplataforma (caracterizarse por funcionar o mantener una interoperabilidad de forma similar en diferentes sistemas operativos).

RNF 9: Legales: La plataforma a utilizar para el desarrollo de la aplicación está basada en la licencia GNU/GPL.

2.5. Definición de los Actores del Sistema

Se definieron como actores del sistema los siguientes:

Tabla 1 Actores del sistema

Actor	Descripción
Creador	Rol que posee los permisos necesarios para gestionar las Salas de Chat, además enviar mensajes privados.
Usuario	Rol que posee los permisos necesarios para autenticarse y enviar mensajes públicos.
Administrador	Rol que posee los permisos necesarios para gestionar el módulo.

Capítulo 2: Análisis y Diseño de un módulo para la integración de Moodle con un sistema de mensajería instantánea.

2.6 Diagrama de Casos de Uso del Sistema

Los casos de uso (CU) son fragmentos de funcionalidad que el sistema ofrece para aportar un resultado de valor para sus actores. De manera más precisa, un CU especifica una secuencia de acciones que el sistema puede llevar a cabo interactuando con sus actores, incluyendo alternativas dentro de la secuencia.

Por tanto, un CU es una especificación. Especifica el comportamiento de cosas dinámicas, en este caso, de instancias de los casos de usos. (Jacobson, y otros, 2004)

En la figura 2 se muestra el diagrama de casos de uso del sistema para la creación de un módulo, donde el creador puede gestionar las Salas de Chat, además de enviar mensajes privados, el Administrador puede gestionar el Módulo y por su parte, el usuario solo puede autenticarse y enviar mensajes públicos.

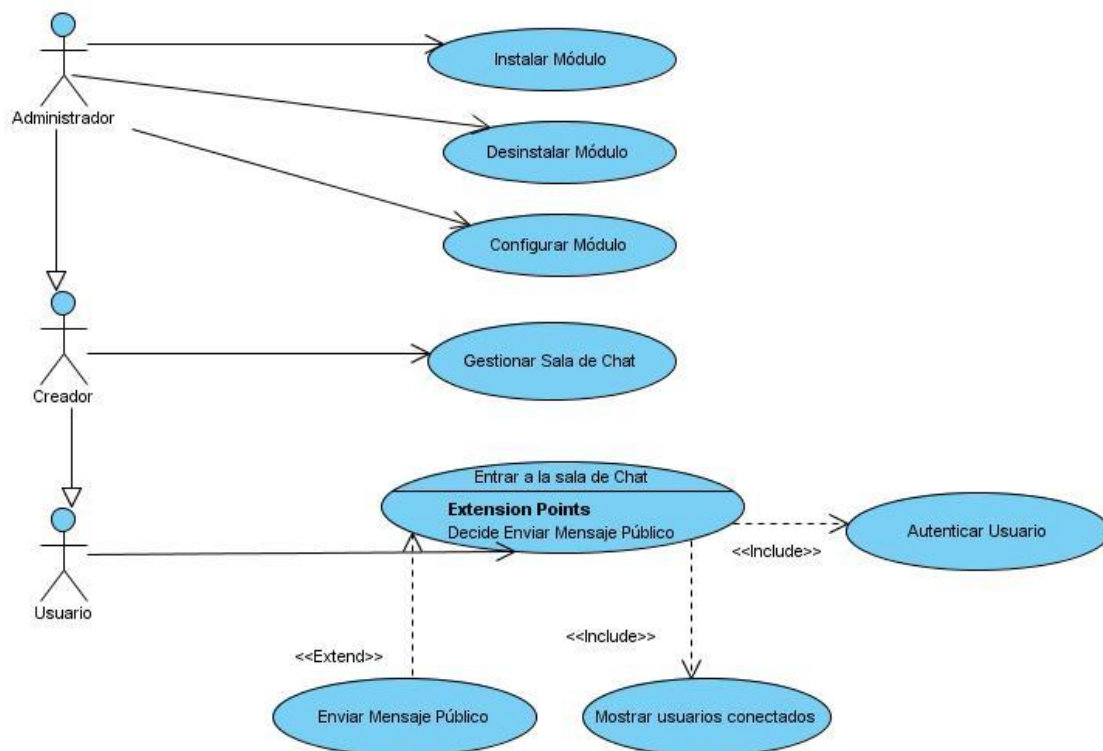


Figura 2 Diagrama de CU del sistema

2.7 Descripción Textual de los Casos de Uso

A continuación se realizan las descripciones textuales de los casos de uso que componen la creación de un módulo para la integración de Moodle con un sistema de mensajería instantánea, con el objetivo de especificar las funcionalidades que contienen.

Capítulo 2: Análisis y Diseño de un módulo para la integración de Moodle con un sistema de mensajería instantánea.

Tabla 2 Descripción del CU Instalar Módulo

Caso de Uso:	Instalar Módulo	
Actores:	Administrador (Inicia)	
Resumen:	El caso de uso inicia cuando el administrador solicita instalar un módulo en Moodle. El sistema realiza la configuración necesaria y finaliza el CU.	
Precondiciones:	Usuario autenticado. La carpeta del módulo debe haberse copiado a la carpeta mod del directorio del Moodle.	
Referencias	RF 1	
Prioridad	Crítico	
Sección “Instalar Módulo”		
Flujo Normal de Eventos		
Acción del Actor	Respuesta del Sistema	
1. El administrador accede a la opción Notificaciones del panel de administración de Moodle.	2. Verifica si existe algún módulo por instalar.	
	3. Almacena la información del módulo.	
	4. Muestra el informe del éxito de la creación de la base de datos del módulo.	
5. Selecciona la opción Continuar.	6. Incorpora el módulo en el listado de módulos reconocidos en Moodle y muestra un informe de éxito de la instalación y finaliza el caso de uso.	
Flujo Alternativo 3a “Fracaso al almacenar información del módulo”		
Acción del Actor	Respuesta del Sistema	
	3a.1. Muestra un informe de los errores ocurridos en el proceso de creación de la base de datos y regresa al paso 1 del Flujo Normal de Eventos.	

Capítulo 2: Análisis y Diseño de un módulo para la integración de Moodle con un sistema de mensajería instantánea.

Poscondiciones	Módulo instalado correctamente.
-----------------------	---------------------------------

Tabla 3 Descripción del CU Configurar Módulo

Caso de Uso:	Configurar Módulo	
Actores:	Administrador (Inicia)	
Resumen:	El caso de uso inicia cuando el Administrador solicita configurar un módulo en Moodle. El sistema realiza la configuración necesaria y finaliza el CU.	
Precondiciones:	Usuario autenticado. Módulo instalado.	
Referencias	RF 2	
Prioridad	Crítico	
Sección “Configurar Módulo”		
Flujo Normal de Eventos		
Acción del Actor	Respuesta del Sistema	
1. El Administrador solicita la opción configurar módulo.	2. Solicita los datos siguientes: <ul style="list-style-type: none"> • Nombre del servidor • HTTP base 	
3. Introduce los datos solicitados.	4. Almacena la información y muestra el mensaje “Cambios actualizados”	
Poscondiciones	Módulo configurado correctamente.	

Tabla 4 Descripción del CU Desinstalar Módulo

Caso de Uso:	Desinstalar Módulo	
Actores:	Administrador (Inicia)	
Resumen:	El caso de uso inicia cuando el Administrador solicita desinstalar un módulo en Moodle. El sistema realiza la configuración necesaria y finaliza el CU.	
Precondiciones:	Usuario autenticado. Módulo instalado.	
Referencias	RF 3	
Prioridad	Crítico	
Sección “Desinstalar Módulo”		

Capítulo 2: Análisis y Diseño de un módulo para la integración de Moodle con un sistema de mensajería instantánea.

Flujo Normal de Eventos	
Acción del Actor	Respuesta del Sistema
1. El Administrador selecciona la opción Borrar módulo.	2. Muestra el siguiente mensaje: “Está a punto de borrar completamente el módulo 'ChatXMMP'. Esto borrará de la base de datos todo lo asociado con este módulo. ¿Está SEGURO de que desea continuar?” Y permite seleccionar la opción: <ul style="list-style-type: none"> • Sí • NO
3. Selecciona la opción SÍ.	4. Elimina toda la información del módulo de la base de datos de Moodle.
	5. Muestra el siguiente mensaje “Todos los datos asociados con el módulo 'ChatXMPP' se han borrado de la base de datos. Para completar el proceso (y prevenir que el módulo se reinstale a sí mismo) debe borrar ahora el directorio <code>/var/www/moodle/mod/Chatxmpp</code> de su servidor”.
	6. Permite seleccionar la opción Continuar y finaliza el CU.
Flujos Alternos	
Flujo Alterno 2a “Selecciona la opción No”	
Acción del Actor	Respuesta del Sistema
	2a.1.Cancela la operación y regresa al paso 1 del Flujo Normal de Eventos de

Capítulo 2: Análisis y Diseño de un módulo para la integración de Moodle con un sistema de mensajería instantánea.

	la Sección “Desinstalar Módulo”.
Poscondiciones	Módulo desinstalado correctamente.

Tabla 5 Descripción del CU Gestionar Sala de Chat excepto Ver

Caso de Uso:	Gestionar Sala de Chat excepto Ver	
Actores:	Creador (Inicia)	
Resumen:	El caso de uso se inicia cuando el Creador necesita adicionar, modificar o eliminar una Sala de Chat. Finaliza el CU.	
Precondiciones:	El Creador debe autenticarse previamente en el sistema.	
Referencias	RF 4	
Prioridad	Crítico	
Flujo Normal de Eventos		
Sección” Adicionar Sala de chat”		
Acción del Actor	Respuesta del Sistema	
El Creador selecciona la opción Agregar Actividad (ChatXMPP).	<ol style="list-style-type: none"> Solicita los siguientes campos: <ul style="list-style-type: none"> Nombre: Solicita el nombre de la sala de chat. Descripción: Breve descripción de la actividad. Tiempo de Inicio: Tiempo que va a iniciar la actividad. Tiempo de Fin: Tiempo en que finaliza la actividad. Modo de grupo: Define si la actividad será accesible para todos los grupos o un grupo en específico. Visible: Especifica si la actividad estará visible u oculta. Número de ID: Número de ID que proporciona una manera de identificar la actividad para los 	

Capítulo 2: Análisis y Diseño de un módulo para la integración de Moodle con un sistema de mensajería instantánea.

	<p>propósitos de cálculo de calidad.</p> <p>Y permite:</p> <ul style="list-style-type: none"> • Guardar cambios y regresar al curso: Almacena la información y regresa al curso. • Guardar cambios y mostrar: Almacena los datos y muestra un vínculo para acceder a la sala. • Cancelar: Cancela la operación.
2. Introduce los datos solicitados y selecciona la opción Guardar cambios y regresar al curso.	3. Comprueba los datos de los campos Nombre, Descripción y Tiempo de Fin.
	4. Almacena los datos y crea la actividad. Finaliza el CU.
Flujos Alternos	
Sección 2a “Selecciona la opción Guardar cambios y mostrar”	
Acción del Actor	Respuesta del Sistema
	2a.1. Almacena los datos y muestra la actividad creada. Finaliza el CU.
Flujos Alternos	
Sección 2b “Selecciona la opción Cancelar”	
Acción del Actor	Respuesta del Sistema
	2. b.1 Cancela la operación. Finaliza el CU.
Flujos Alternos	
Sección 4a “Campos vacíos”	
Acción del Actor	Respuesta del Sistema
	4a.1. Emite un mensaje de error sobre el/los campos sin llenar: “Debe suministrar un valor aquí”.
Flujos Alternos	
Sección 4b “Campos incorrectos”	
Acción del Actor	Respuesta del Sistema

Capítulo 2: Análisis y Diseño de un módulo para la integración de Moodle con un sistema de mensajería instantánea.

	4.b.1. Emite un mensaje de error sobre el campo Tiempo de Fin: “La fecha de fin debe ser mayor que la de inicio”.
Sección “ Modificar Sala de chat”	
Acción del Actor	Respuesta del Sistema
1. Selecciona la opción Modificar ChatXMPP.	<p>2. Muestra los campos a modificar:</p> <ul style="list-style-type: none"> • Nombre: Solicita el nombre de la sala de chat. • Descripción: Breve descripción de la actividad. • Tiempo de Inicio: Tiempo que va a iniciar la actividad. • Tiempo de Fin: Tiempo en que finaliza la actividad. • Modo de grupo: Define si la actividad será accesible para todos los grupos o un grupo en específico. • Visible: Especifica si la actividad estará visible u oculta. • Número de ID: Número de ID que proporciona una manera de identificar la actividad para los propósitos de cálculo de calidad. <p>Y permite:</p> <ul style="list-style-type: none"> • Guardar cambios y regresar al curso: Almacena la información y regresa al curso. • Guardar cambios y mostrar: Almacena los datos y muestra un vínculo para acceder a la sala. • Cancelar: Cancela la operación.

Capítulo 2: Análisis y Diseño de un módulo para la integración de Moodle con un sistema de mensajería instantánea.

3. Introduce los datos solicitados.	4. Comprueba los datos de los campos Nombre, Descripción y Tiempo de Fin.
	5. Almacena los datos y crea la actividad. Finaliza el CU.
Flujos Alternos	
Sección 2a “Selecciona la opción Guardar cambios y mostrar”	
Acción del Actor	Respuesta del Sistema
	2a.1. Almacena los datos y muestra la actividad creada. Finaliza el CU.
Flujos Alternos	
Sección 2b “Selecciona la opción Cancelar”	
Acción del Actor	Respuesta del Sistema
	2b.1 Cancela la operación. Finaliza el CU.
Flujos Alternos	
Sección 4a “Campos vacíos”	
Acción del Actor	Respuesta del Sistema
	4a.1. Emite un mensaje de error sobre el/los campos sin llenar: “Debe suministrar un valor aquí”.
Flujos Alternos	
Sección 4b “Campos incorrectos”	
Acción del Actor	Respuesta del Sistema
	4b.1. Emite un mensaje de error sobre el campo Tiempo de Fin: “La fecha de fin debe ser mayor que la de inicio”.
Sección” Eliminar Sala de chat”	
Acción del Actor	Respuesta del Sistema
1. Selecciona la opción Eliminar ChatXMPP.	2. Muestra el siguiente mensaje de confirmación: “¿Está seguro que desea borrar definitivamente el Chat?” Y permite seleccionar la opción:

Capítulo 2: Análisis y Diseño de un módulo para la integración de Moodle con un sistema de mensajería instantánea.

	<ul style="list-style-type: none"> • Sí • NO
3. Selecciona la opción Sí.	4. Elimina toda la información del ChatXMPP.
Flujos Alternos	
Flujo Alterno 2a “Selecciona la opción No”	
Acción del Actor	Respuesta del Sistema
	4a.1. Se cancela la operación y termina el CU.
Poscondiciones	Se adiciona, se modifica y/o elimina una Sala de chat.

Tabla 6 Descripción CU Autenticar Usuario

Caso de Uso:	Autenticar Usuario
Actores:	Usuario
Resumen:	El caso de uso se inicia cuando el Usuario está dentro de la plataforma Moodle y desea entrar a la Sala de Chat. Finaliza el CU.
Precondiciones:	El Usuario debe tener nombre de usuario y contraseña.
Referencias	RF 5
Prioridad	Crítico
Flujo Normal de Eventos	
Sección “Autenticar Usuarios”	
Acción del Actor	Respuesta del Sistema
1. Solicita el servicio de autenticación.	2. Muestra la opción para realizar la autenticación del usuario, donde debe recoger los siguientes datos: <ul style="list-style-type: none"> • Usuario. • Contraseña. Y permite seleccionar la opción Entrar.
3. Introduce los datos y selecciona la	4. Valida los datos.

Capítulo 2: Análisis y Diseño de un módulo para la integración de Moodle con un sistema de mensajería instantánea.

opción de entrar.	
	5. Actualiza en la base de datos los datos del usuario autenticado.
Flujos Alternos	
Sección 4b “Fallo en la autenticación”	
Acción del Actor	Respuesta del Sistema
	4b.2. Muestra un mensaje de error que falló la autenticación.
Poscondiciones	Se autentica el Usuario de manera satisfactoria.

Tabla 7 Descripción del CU Entrar a la Sala de Chat

Caso de Uso:	Entrar a la Sala de Chat	
Actores:	Usuario	
Resumen:	El caso de uso se inicia cuando el Usuario selecciona la Sala de chat para entrar en la misma y este se autentica. Finaliza el CU.	
Precondiciones:	La Sala de Chat debe estar creada.	
Referencias	RF 6	
Prioridad	Crítico	
Flujo Normal de Eventos		
Sección “Entrar a la sala de Chat”		
Acción del Actor	Respuesta del Sistema	
1. El Usuario selecciona la opción Entra a la sala.	2. Muestra la ventana de la sala de Chat, los usuarios que se encuentran conectados y los mensajes enviados.	
Flujos Alternos		
Flujo Alterno 3a “Entrar a la Sala de Chat fuera de fecha”		
Acción del Actor	Respuesta del Sistema	
	3a.2. El sistema muestra un mensaje de error “No puede seleccionar esta Sala de Chat porque está fuera de fecha”. Y	

Capítulo 2: Análisis y Diseño de un módulo para la integración de Moodle con un sistema de mensajería instantánea.

	muestra el botón Continuar.
Poscondiciones	Se entra en la Sala de Chat.

Tabla 8 Descripción del CU Mostrar Listado de Usuarios Conectados

Caso de Uso:	Mostrar Listado de Usuarios Conectados	
Actores:	Usuario	
Resumen:	El caso de uso se inicia cuando el usuario está autenticado dentro de la Sala de chat y se muestran los usuarios conectados. Finaliza el CU.	
Precondiciones:	El usuario debe estar dentro de la Sala de chat.	
Referencias	RF 7	
Prioridad	Crítico	
Flujo Normal de Eventos		
Sección “Mostrar Usuarios Conectados”		
	Acción del Actor	Respuesta del Sistema
		1. Muestra los usuarios conectados en la Sala de Chat.
Poscondiciones	Se muestran los usuarios conectados a la Sala de Chat	

Tabla 9 Descripción del CU Enviar Mensaje Público

Caso de Uso:	Enviar Mensaje Público	
Actores:	Usuario	
Resumen:	El caso de uso se inicia cuando un Usuario está dentro de la Sala de chat y envía un mensaje público. Finaliza el CU.	
Precondiciones:	El Usuario debe estar dentro de la Sala de chat.	
Referencias	RF 8	
Prioridad	Crítico	
Flujo Normal de Eventos		
Sección “Enviar Mensaje Público”		
	Acción del Actor	Respuesta del Sistema
	1. El Usuario introduce el mensaje de texto y	2. Muestra el mensaje enviado y

Capítulo 2: Análisis y Diseño de un módulo para la integración de Moodle con un sistema de mensajería instantánea.

pulsa la opción Enviar mensaje público.	termina el CU.
Poscondiciones	Enviado con éxito el mensaje público.

2.8 Modelo de Análisis.

El Modelo de Análisis es una especificación detallada de los requisitos. En él se identifican una serie de clases y relaciones que relacionadas entre sí permiten la interpretación del funcionamiento de la parte del sistema que se esté modelando. Este modelo funciona como una primera aproximación del modelo de diseño. A continuación se muestran los diagramas de análisis por casos de uso correspondientes al módulo Chat sobre protocolo XMPP.

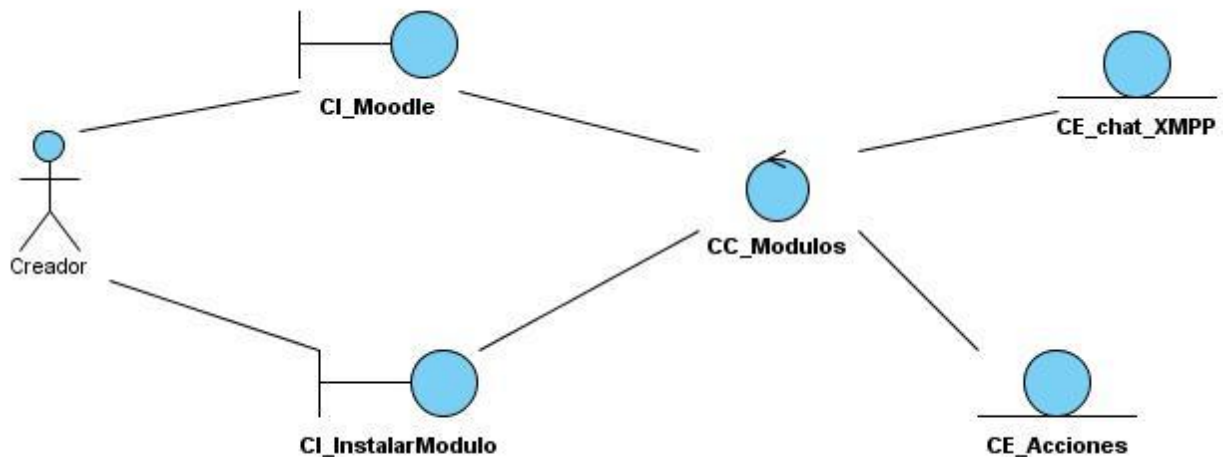


Figura 3 Diagrama de clases del análisis CU Instalar Módulo.

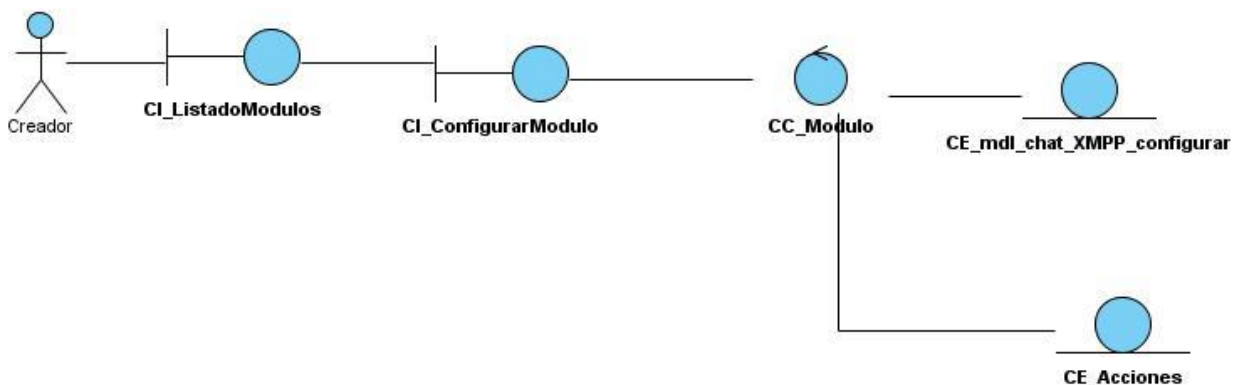


Figura 4 Diagrama de clases del análisis del CU Configurar Módulo.

Capítulo 2: Análisis y Diseño de un módulo para la integración de Moodle con un sistema de mensajería instantánea.

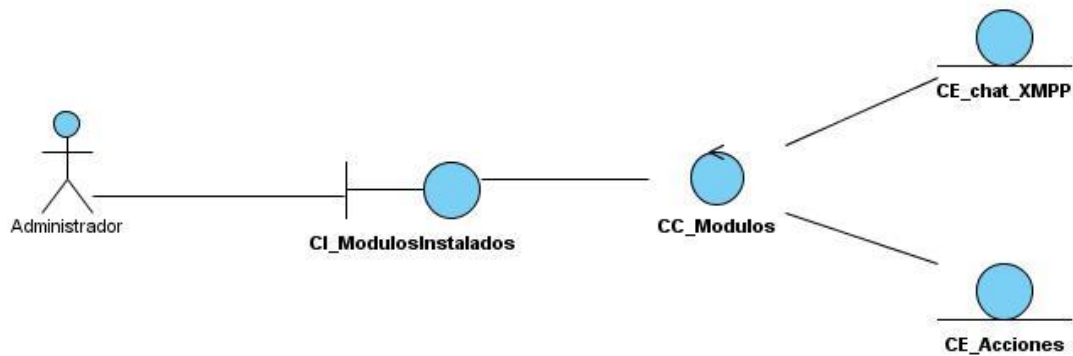


Figura 5 Diagrama de clases del análisis del CU Desinstalar Módulo.

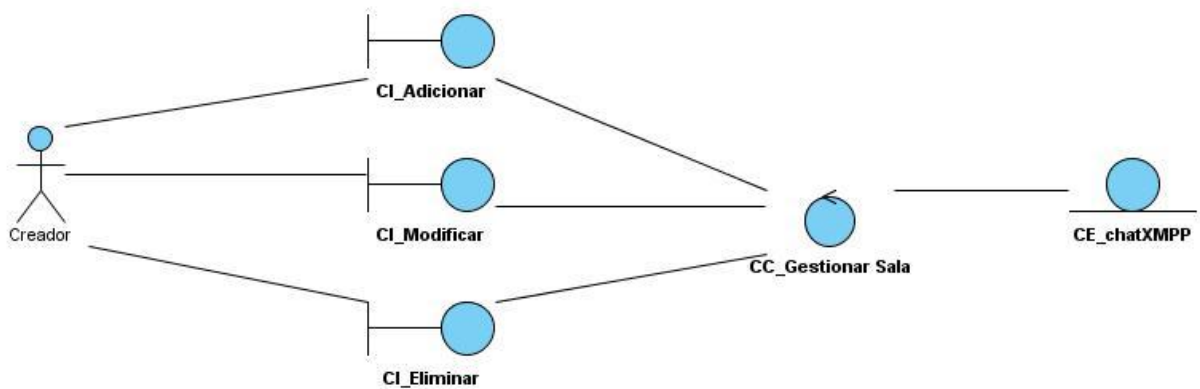


Figura 6 Diagrama de clases del análisis del CU Gestionar Sala de Chat.

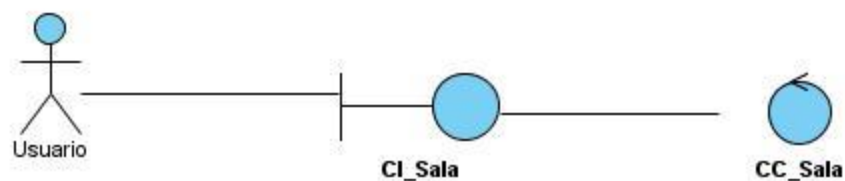


Figura 7 Diagrama de clases del análisis del CU Autenticar Usuario.

Capítulo 2: Análisis y Diseño de un módulo para la integración de Moodle con un sistema de mensajería instantánea.

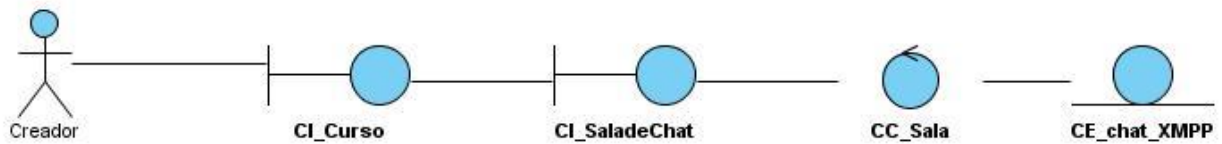


Figura 8 Diagrama de clases del análisis del CU Entrar a la Sala de Chat.

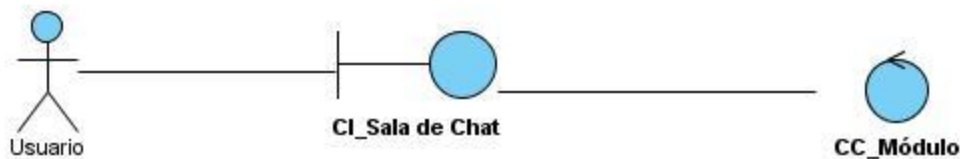


Figura 9 Diagrama de clases del análisis del CU Mostrar Usuarios Conectados.



Figura 10 Diagrama de clases del análisis del CU Enviar Mensaje Público.

2.9 Diagramas de Interacción

Los diagramas de interacción constituyen una de las herramientas más importantes para el análisis y diseño orientado a objetos, ya que explican gráficamente cómo los objetos interactúan a través de mensajes para realizar las tareas, estas interacciones se pueden representar a través de diagramas de secuencia o de colaboración. Para la solución que se propone, se emplean diagramas de colaboración, pues el objetivo fundamental es identificar requisitos y responsabilidades sobre los objetos.

2.9.1 Diagramas de Colaboración

Los diagramas de colaboración representan el comportamiento de los objetos en tiempo de ejecución, por medio de paso de mensajes.

Capítulo 2: Análisis y Diseño de un módulo para la integración de Moodle con un sistema de mensajería instantánea.

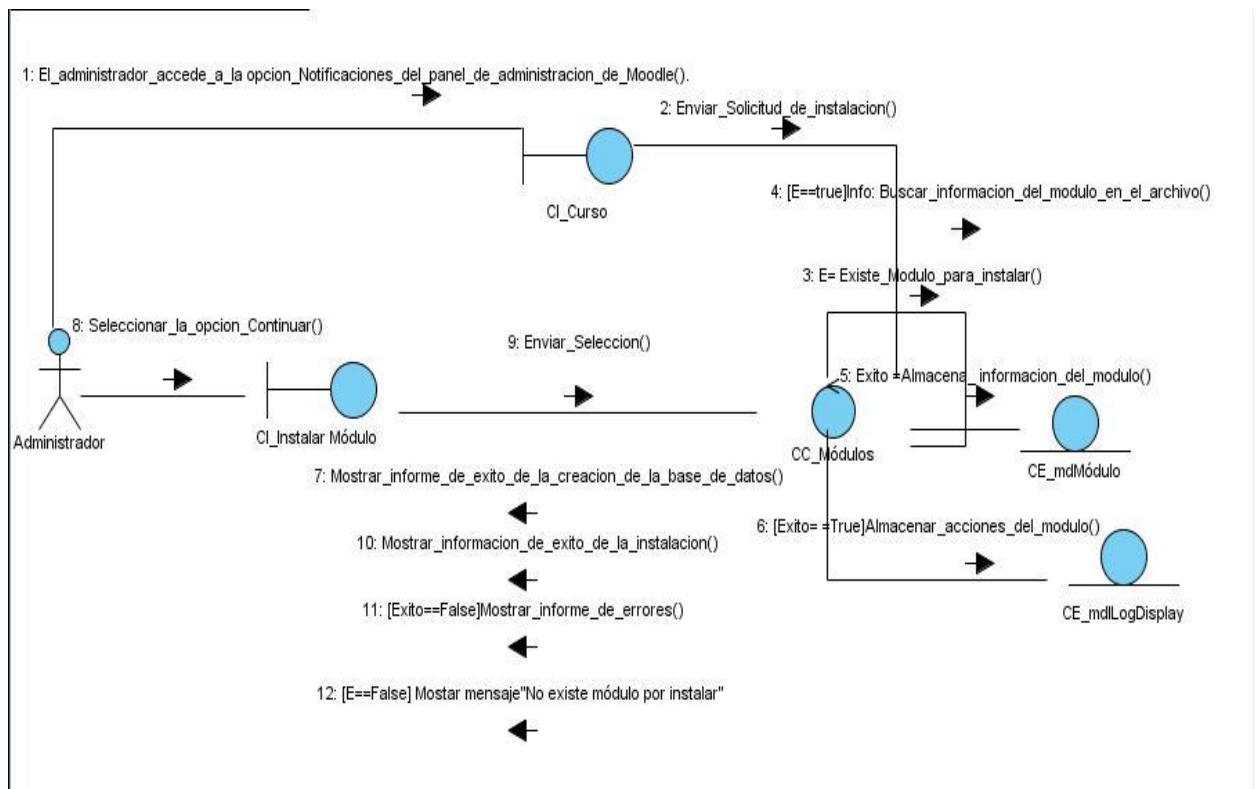


Figura 11 Diagrama de Colaboración del CU Instalar Módulo.

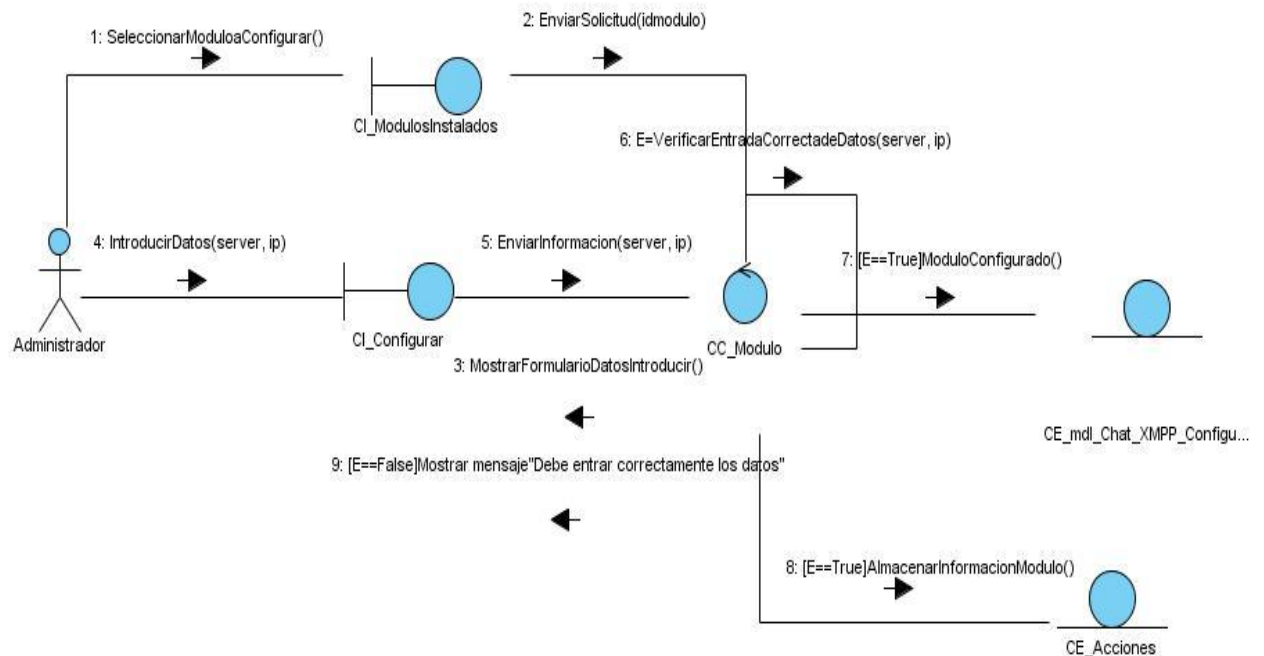


Figura 12 Diagrama de Colaboración del CU Configurar Módulo.

Capítulo 2: Análisis y Diseño de un módulo para la integración de Moodle con un sistema de mensajería instantánea.

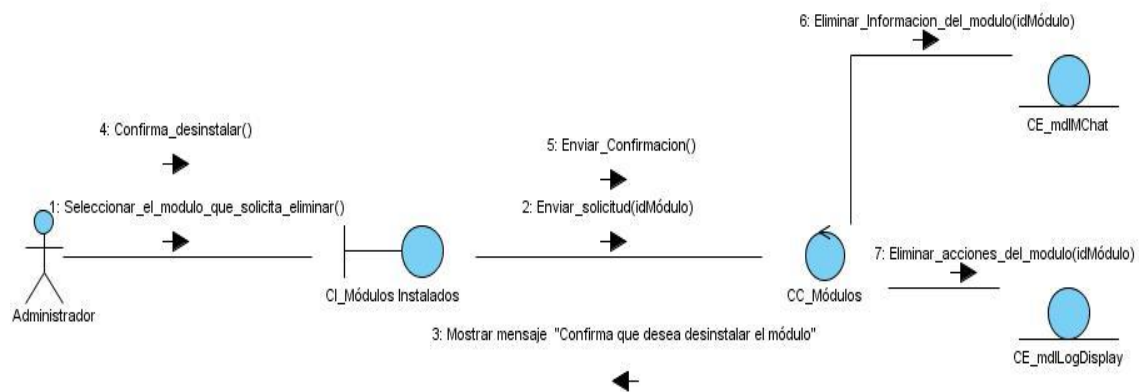


Figura 13 Diagrama de Colaboración del CU Desinstalar Módulo.

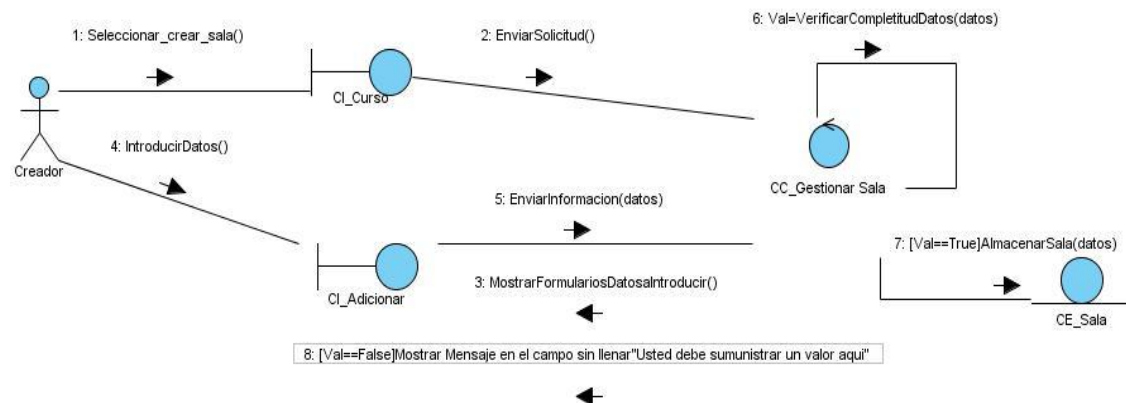


Figura 14 Diagrama de Colaboración del CU Adicionar Sala de Chat.

Capítulo 2: Análisis y Diseño de un módulo para la integración de Moodle con un sistema de mensajería instantánea.

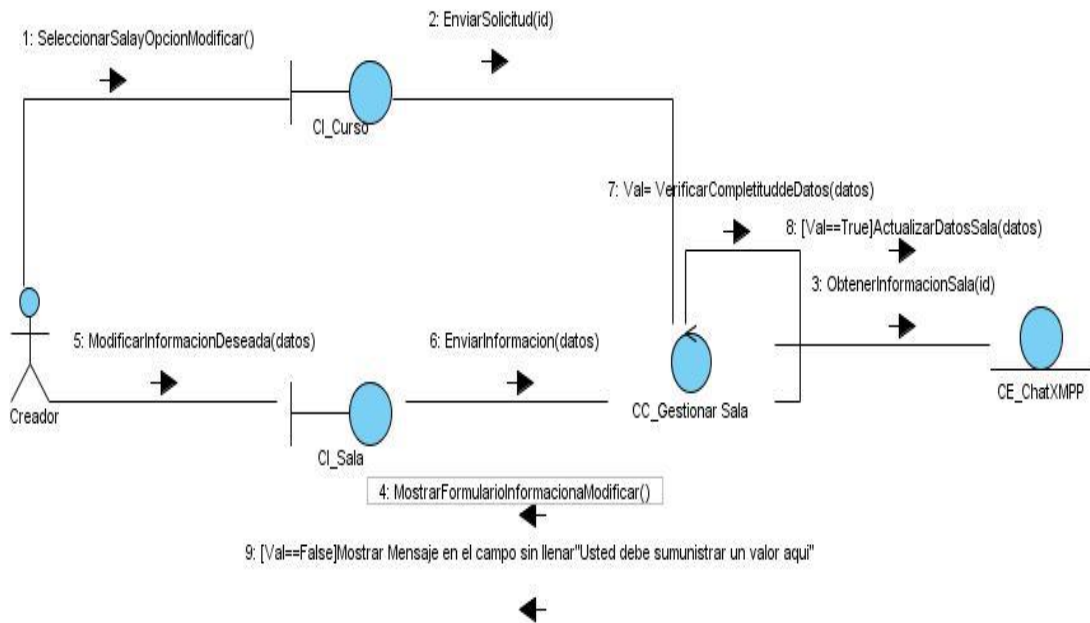


Figura 15 Diagrama de Colaboración del CU Modificar Sala de Chat.

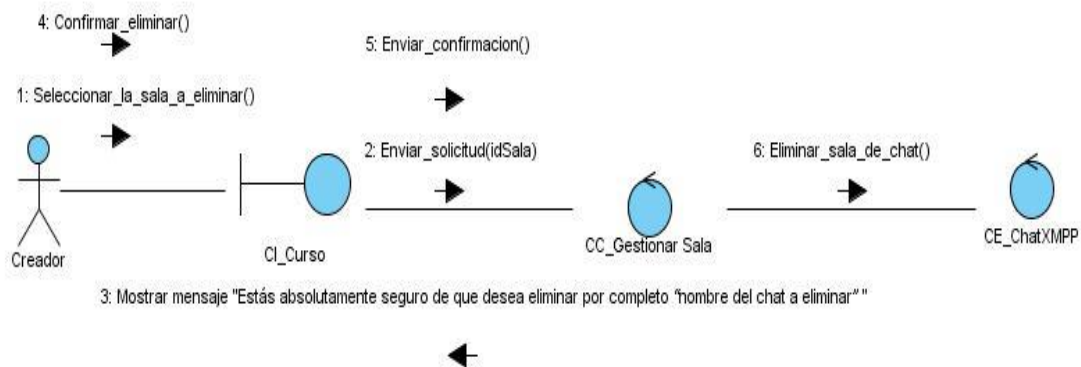


Figura 16 Diagrama de Colaboración del CU Eliminar Sala de Chat.

Capítulo 2: Análisis y Diseño de un módulo para la integración de Moodle con un sistema de mensajería instantánea.

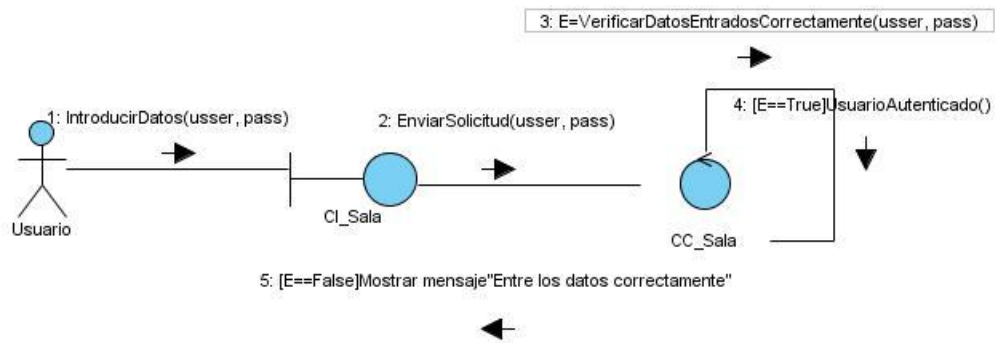


Figura 17 Diagrama de Colaboración del CU Autenticar Usuario.

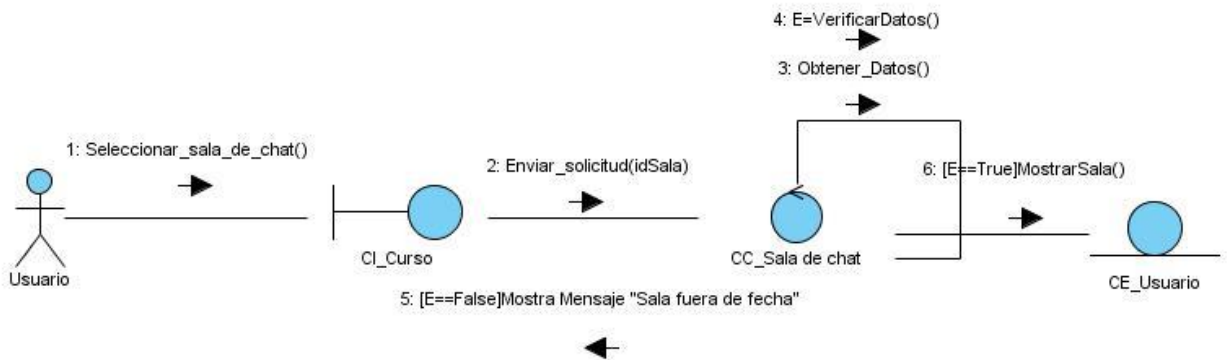


Figura 18 Diagrama de Colaboración del CU Entrar a la Sala de Chat.

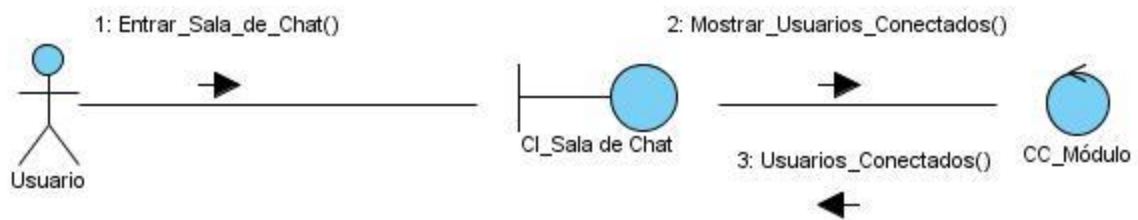


Figura 19 Diagrama de Colaboración del CU Mostrar Usuarios Conectados.

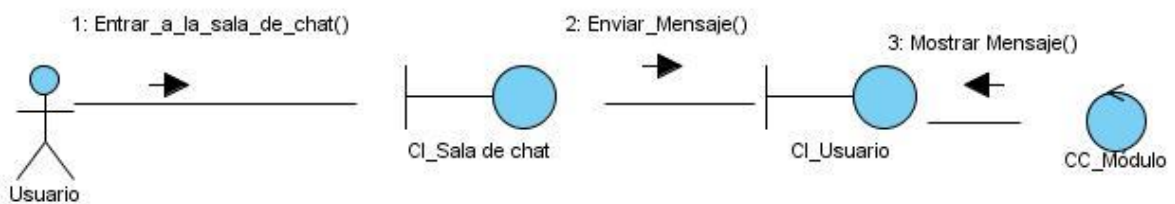


Figura 20 Diagrama de Colaboración del CU Enviar Mensaje Público.

Capítulo 2: Análisis y Diseño de un módulo para la integración de Moodle con un sistema de mensajería instantánea.

2.10 Modelo de Diseño

El Modelo de Diseño es el modelo de objetos que permite la descripción física de los casos de uso, centrándose en los requisitos funcionales y no funcionales, junto con otras restricciones relacionadas con el entorno de implementación, tienen impacto en el sistema. Este artefacto constituye la entrada fundamental para el correcto desarrollo de las entradas de implementación. A continuación se presentan los diagramas de diseño Web correspondientes a cada caso de uso:

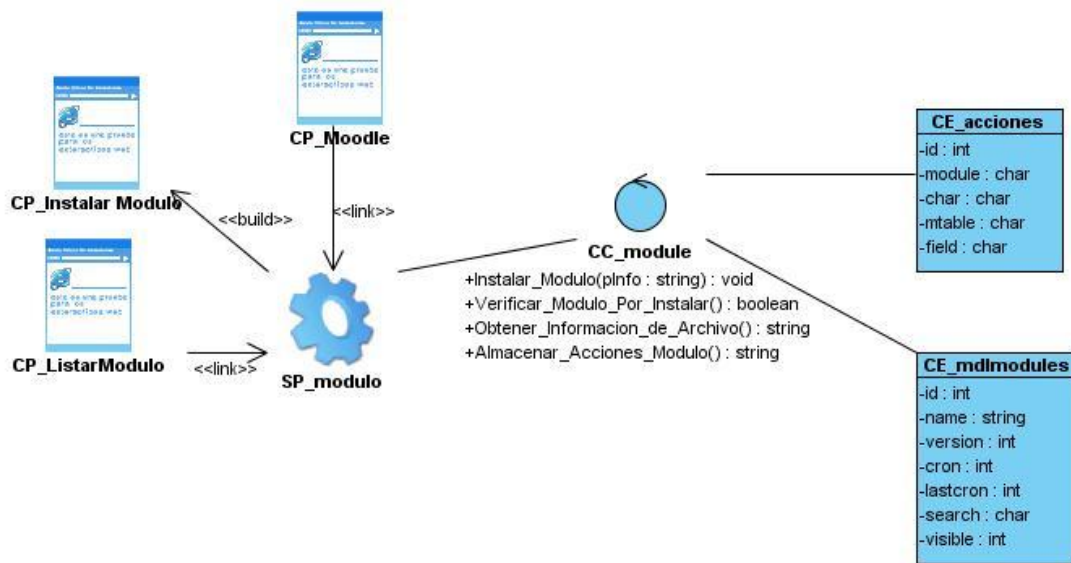


Figura 21: Diagrama de Clases del Diseño del CU Instalar Módulo.

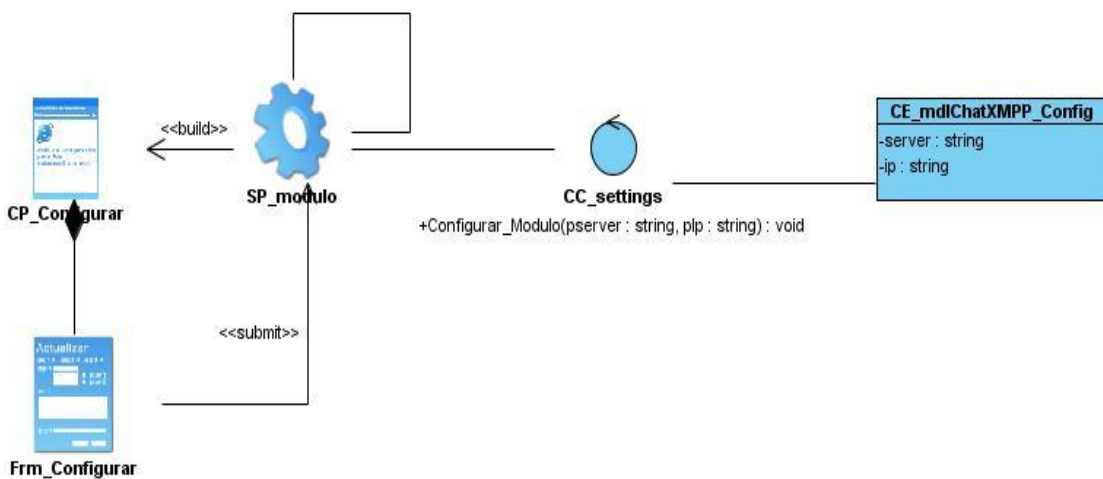


Figura 22: Diagrama de Clases del Diseño del CU Configurar Módulo.

Capítulo 2: Análisis y Diseño de un módulo para la integración de Moodle con un sistema de mensajería instantánea.

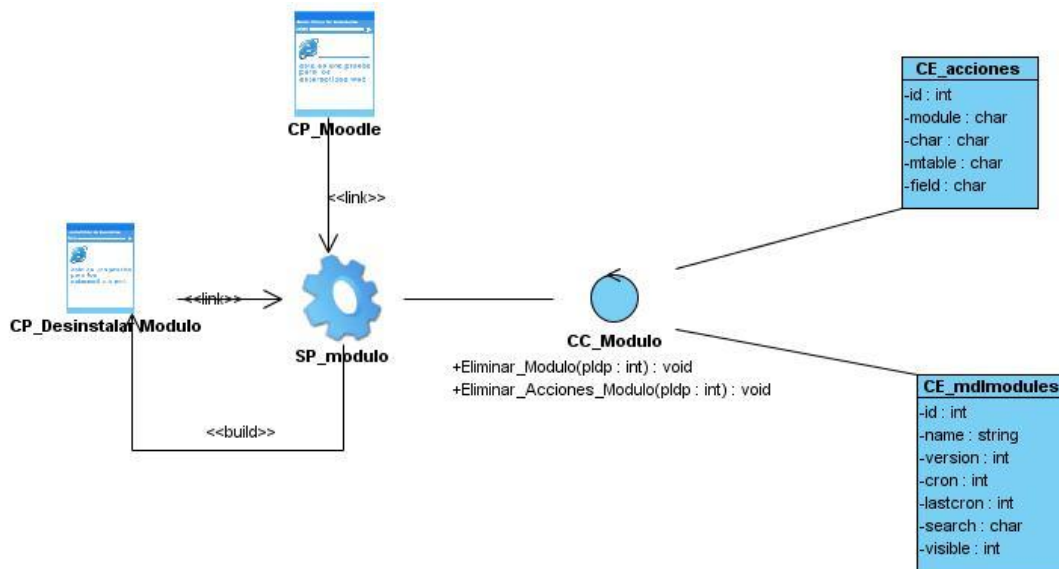


Figura 23: Diagrama de Clases del Diseño del CU Desinstalar Módulo.

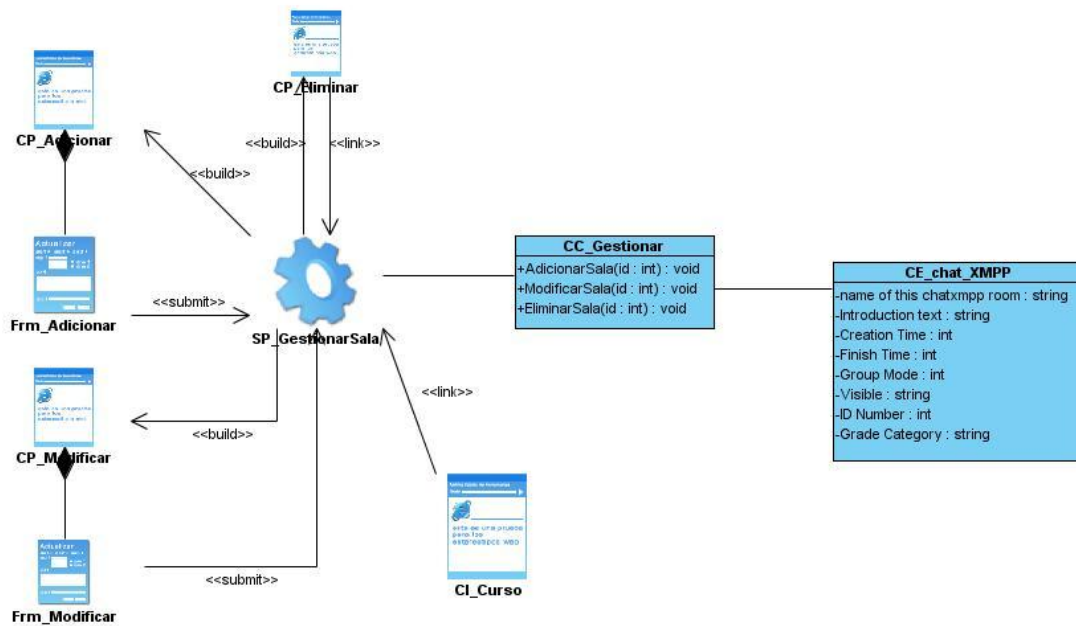


Figura 24: Diagrama de Clases del Diseño del CU Gestionar Sala de Chat.

Capítulo 2: Análisis y Diseño de un módulo para la integración de Moodle con un sistema de mensajería instantánea.

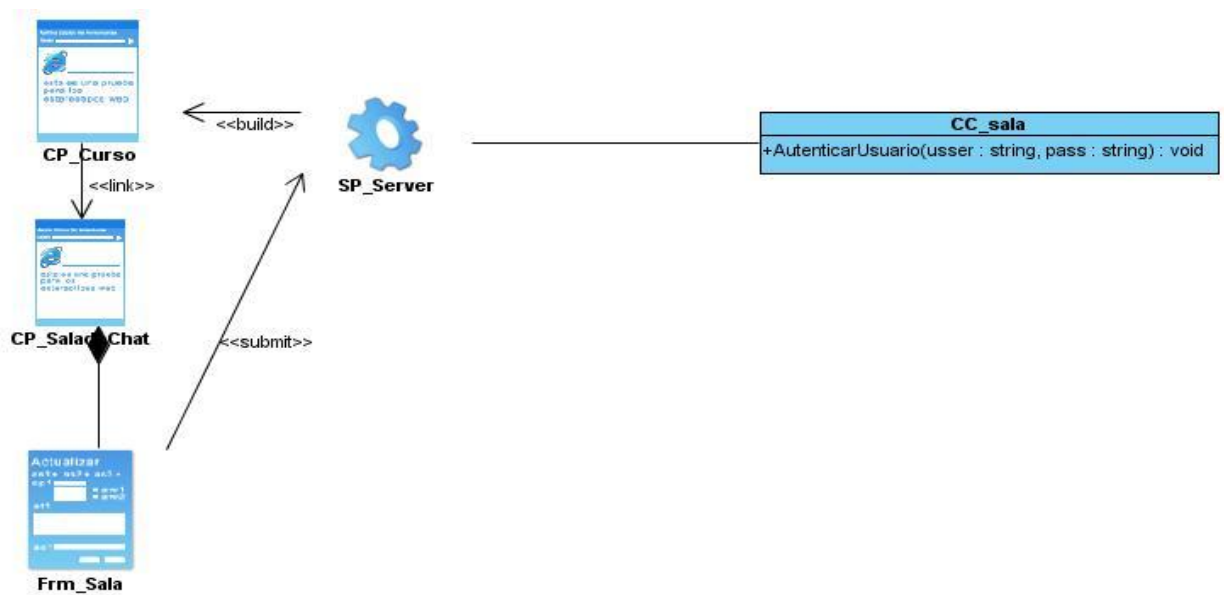


Figura 25: Diagrama de Clases del Diseño del CU Autenticar Usuario.

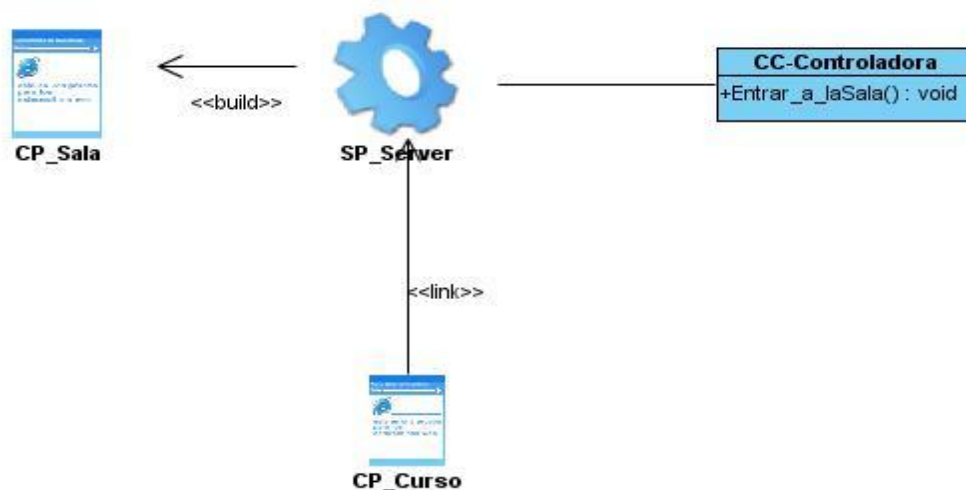


Figura 26: Diagrama de Clases del Diseño del CU Entrar a la Sala de Chat.

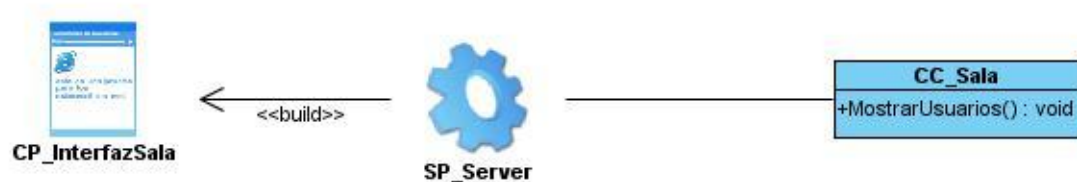


Figura 27: Diagrama de Clases del Diseño del CU Mostrar Usuarios Conectados.

Capítulo 2: Análisis y Diseño de un módulo para la integración de Moodle con un sistema de mensajería instantánea.

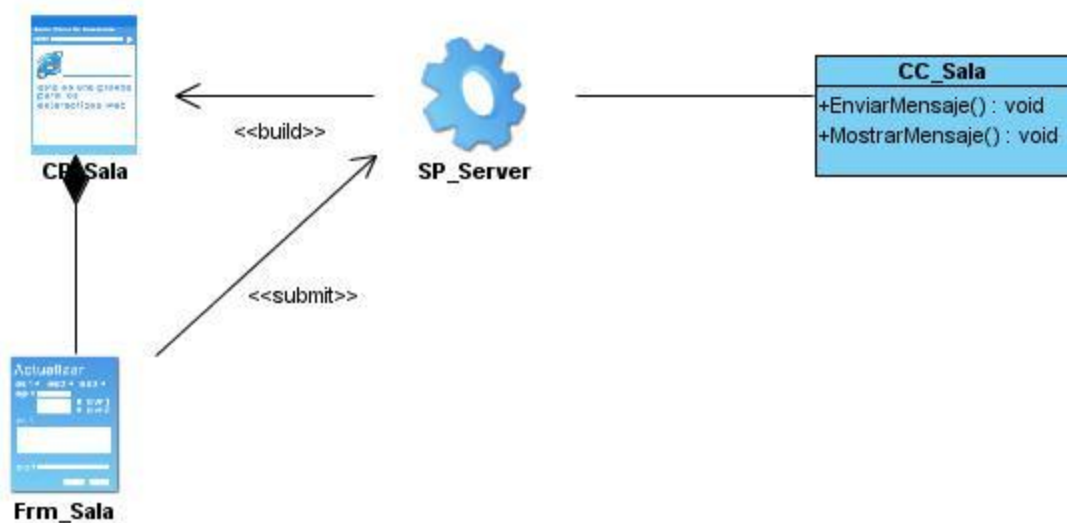


Figura 28: Diagrama de Clases del Diseño del CU Enviar Mensaje Público.

2.11 Modelo de Datos

El Modelo de Datos permite la descripción de los elementos que intervienen en un problema dado y la forma en que se relacionan entre sí.

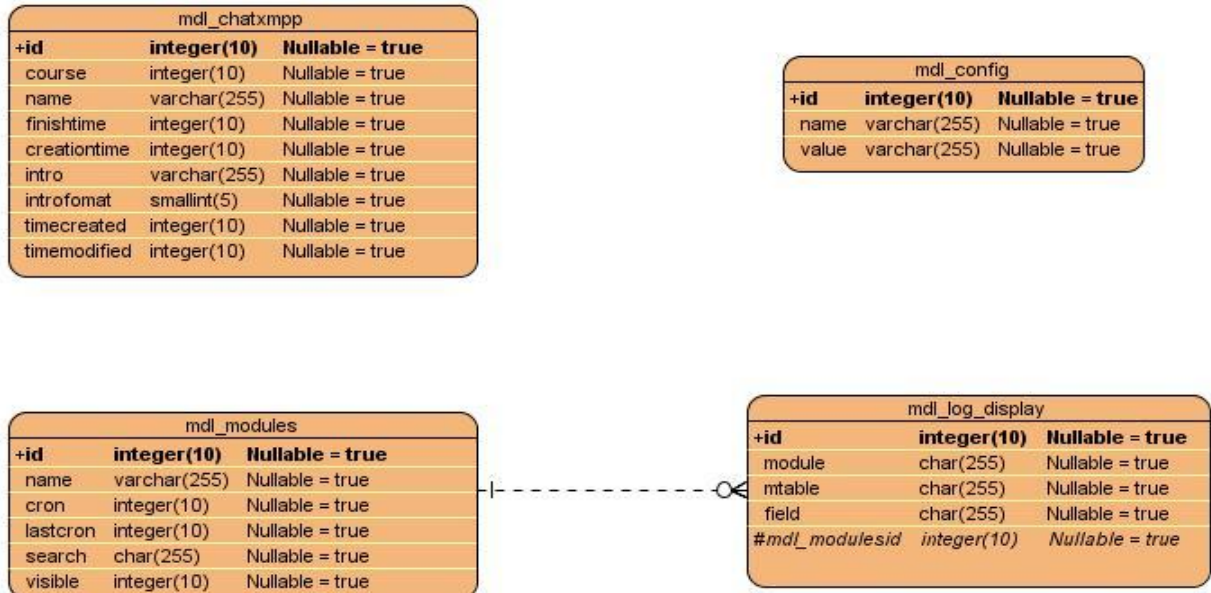


Figura 29: Modelo de Datos.

Capítulo 2: Análisis y Diseño de un módulo para la integración de Moodle con un sistema de mensajería instantánea.

Conclusiones parciales

Con el desarrollo del capítulo se llegaron a las siguientes conclusiones:

- Los artefactos generados en este capítulo son de gran importancia para la investigación, ya que cada uno de ellos tributa a un mejor entendimiento por los implementadores sobre un sistema de mensajería instantánea a desarrollar.
- A partir de la identificación de los requisitos funcionales que debe cumplir el software se pudieron determinar los distintos casos de uso del sistema a desarrollar, los cuales fueron representados en el modelo de casos de uso del sistema y descritos textualmente, garantizando una comprensión exacta de lo que se desea desarrollar.
- Los diagramas de interacción y de diseño web constituyen el punto de partida para la implementación del sistema de mensajería instantánea.

Capítulo 3: Implementación del Módulo ChatXMPP

Introducción

En este capítulo se explica cómo los elementos del diseño se implementan en términos de componentes a través del modelo de implementación, el cual está compuesto por los diagramas de despliegue y de componentes. Estos diagramas describen los componentes a construir, su organización y dependencia entre los nodos físicos en los que funciona la aplicación. Se muestra además, los elementos de la base de datos a través del modelo de datos y una descripción de los casos de prueba para cada caso de uso del sistema.

3.1 Modelo de Implementación

El modelo de implementación describe cómo los elementos del modelo de diseño, como las clases, se implementan en términos de componentes, como ficheros de código fuente, ejecutables, etc. El modelo de implementación describe también cómo se organizan los componentes de acuerdo con los mecanismos de estructuración y modularización disponibles en el entorno de implementación y en el lenguaje o lenguajes de programación utilizados, y cómo dependen los componentes unos de otros. (Jacobson, y otros, 2004)

3.1.1 Diagrama de Componentes

Un diagrama de componentes muestra las organizaciones y dependencias lógicas entre componentes de software, sean estos de código fuente, binarios o ejecutables. A partir de este diagrama se tienen en consideración los requisitos relacionados con la facilidad de desarrollo, la gestión de software, la reutilización, y las restricciones impuestas por los lenguajes de programación y las herramientas utilizadas en el desarrollo.

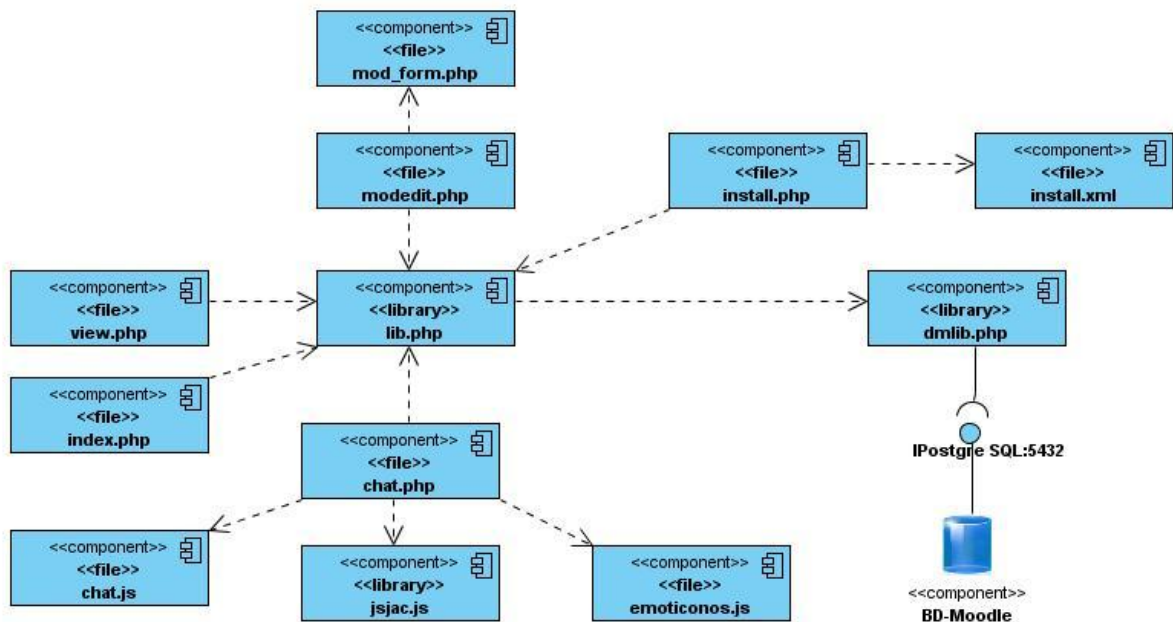


Figura 30: Diagrama de Componentes.

3.1.2 Diagrama de Despliegue

El diagrama de despliegue se utiliza para capturar los elementos de configuración del procesamiento y las conexiones entre estos. Se aplica además, para visualizar la distribución de los componentes de software en los nodos físicos.

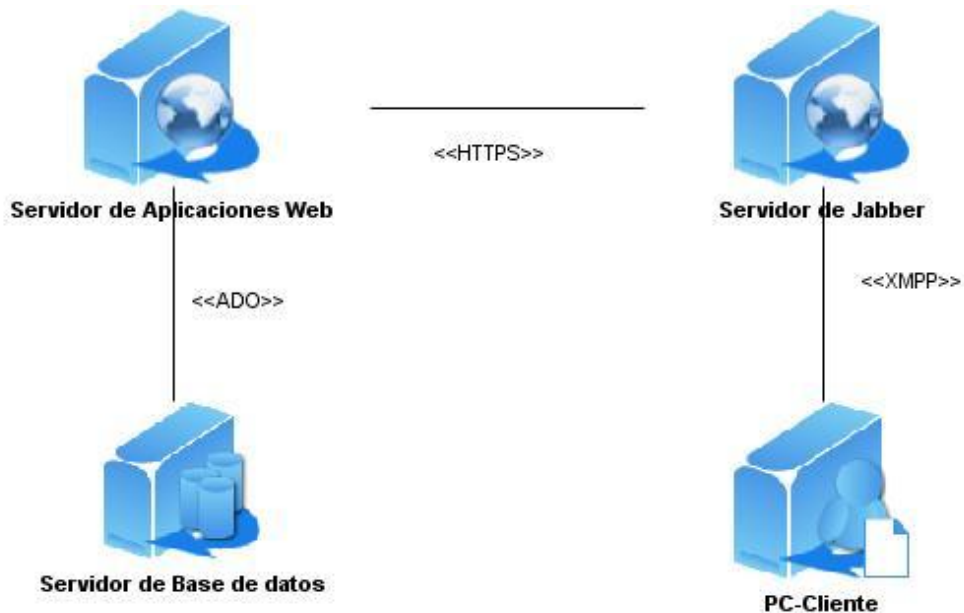


Figura 31: Diagrama de Despliegue.

3.2 Pruebas

Las pruebas constituyen un elemento importante en la garantía de la calidad de un software, ya que permiten evaluar un sistema bajo condiciones o requerimientos específicos, donde sus resultados pueden ser observados y registrados. Es por medio de ellas que se puede asegurar un correcto funcionamiento de un sistema, debido a que permiten identificar y corregir a tiempo los posibles errores que pueda contener un software antes de ser desplegado.

3.2.1 Niveles de Prueba

Entre los niveles de prueba se encuentran el nivel de pruebas unitarias, nivel de pruebas de integración, nivel de pruebas del sistema y nivel de pruebas de aceptación.

- **Pruebas Unitarias:** Comienzan con la prueba de cada módulo. Una prueba unitaria es una forma de probar el correcto funcionamiento de un módulo de código. El objetivo de las pruebas unitarias es aislar cada parte del programa y mostrar que las partes individuales son correctas. Proporcionan un contrato escrito que el fragmento de código debe satisfacer. Estas pruebas aisladas proporcionan cinco ventajas básicas; fomentan el cambio, simplifican la integración, documentan el código, separan la interfaz del código y hacen que los errores estén más acotados y sean fáciles de localizar. (Jacobson, y otros, 2004)
- **Pruebas de Integración:** A partir del esquema del diseño, los módulos probados se vuelven a probar combinados para probar sus interfaces. Pruebas integrales o pruebas de integración son aquellas que se realizan en el ámbito del desarrollo de software una vez que se han aprobado las pruebas unitarias. Únicamente se refieren a la prueba o pruebas de todos los elementos unitarios que componen un proceso, hecha en conjunto, de una sola vez. Consiste en realizar pruebas para verificar que un gran conjunto de partes de software funcionan juntos. (Jacobson, y otros, 2004)
- **Pruebas del Sistema:** El software ensamblado totalmente con cualquier componente hardware que requiera, se prueba para comprobar que se cumplen los requisitos funcionales. Cualquier pieza de software completo, desarrollado o adquirido, puede verse como un sistema que debe probarse, ya sea para decidir acerca de su aceptación, para analizar defectos globales o para estudiar aspectos específicos de su comportamiento, tales como seguridad o rendimiento. Este tipo de pruebas estudia el producto completo. (Jacobson, y otros, 2004)

Capítulo 3: Implementación del Módulo ChatXMPP.

3.2.2 Método de prueba

RUP propone dos métodos fundamentales: Caja Blanca y Caja Negra. A continuación se describen ambos métodos, haciéndose mayor énfasis en las pruebas de Caja Negra ya que serán las más utilizadas en la comprobación de la solución.

Pruebas de Caja Negra: Las pruebas de caja negra, también denominadas pruebas de comportamiento, se centran en los requisitos funcionales del software. O sea, permiten al ingeniero del software obtener conjuntos de condiciones de entrada que ejerciten completamente todos los requisitos funcionales de un programa. Estas pruebas no son una alternativa a las técnicas de pruebas de caja blanca, más bien se trata de un enfoque complementario que intenta descubrir diferentes tipos de errores que se llevan a cabo sobre la interfaz del software. El objetivo es demostrar que las funciones del software son operativas, que las entradas se aceptan de forma adecuada y se produce un resultado correcto, y que la integridad de la información externa se mantiene, saber qué es lo que hace el software pero sin entrar en detalles de código, es decir, que es lo que hace, y no cómo lo hace (no se ve el código).

Estas pruebas permiten encontrar (Pressman, Roger S, 2002):

- Funciones incorrectas o ausentes.
- Errores de interfaz.
- Errores en estructuras de datos o en accesos a las bases de datos externas.
- Errores de rendimiento.
- Errores de inicialización y terminación.

Pruebas de Caja Blanca: La prueba de caja blanca, denominada a veces prueba de caja de cristal es un método de diseño de casos de prueba que usa la estructura de control del diseño procedimental para obtener los casos de prueba. Mediante los métodos de prueba de caja blanca, el ingeniero del software puede obtener casos de prueba que garanticen que se ejercita por lo menos una vez todos los caminos independientes de cada módulo, ejerciten todas las decisiones lógicas en sus vertientes verdadera y falsa; ejecuten todos los ciclos en sus límites y con sus límites operacionales, y ejerciten las estructuras internas de datos para asegurar su validez (Pressman, Roger S, 2002).

3.2.3 Diseño de Casos de Pruebas

Con el propósito de probar las nuevas funcionalidades incorporadas al módulo ChatXMPP, se efectúa el diseño de los casos de pruebas para los casos de uso del sistema, con el fin de detectar aquellos errores que pueda contener la aplicación.

Capítulo 3: Implementación del Módulo ChatXMPP.

Con este objetivo se ejecuta cada proceso o función usando datos válidos y no válidos, para verificar que:

- Se obtengan los resultados esperados cuando se emplean datos válidos.
- Se despliegan mensajes de error o advertencias apropiadas cuando se gestionan datos erróneos.

Capítulo 3: Implementación del Módulo ChatXMPP.

Tabla 10 Caso de Prueba de la Sección Adicionar Sala de Chat

Escenario	Descripción	Nombre	Descripción	Tiempo Inicio	Tiempo Fin	Modo Grupo	Visible	No ID	Respuesta del sistema	Flujo central
E C 1.1	Selecciona la opción agregar actividad: Adicionar ChatXMPP.								<p>Solicita los siguientes campos:</p> <ul style="list-style-type: none"> • Nombre: • Descripción • Tiempo de Inicio: • Tiempo de Fin • Modo de grupo • Visible • Número de ID <p>Y permite:</p> <ul style="list-style-type: none"> • Guardar cambios y regresar al curso • Guardar cambios y mostrar • Cancelar 	Moodle/Agregar Actividad/ChatXMPP

Capítulo 3: Implementación del Módulo ChatXMPP.

EC 1.2	Introduce los datos y selecciona la opción cambios y regresar al curso.	V	V	N/A	V	N/A	N/A	N/A	<ol style="list-style-type: none"> 1. Comprueba los datos de los campos Nombre, Descripción y Tiempo de Fin. 2. Almacena los datos y crea la actividad. 	Moodle/Agregar Actividad/ChatXMPP /Botón Salvar los cambios y regresar al curso.
EC 1.3	Selecciona la opción Guardar cambios y mostrar								Almacena los datos y muestra la actividad creada.	Moodle/Agregar Actividad/ChatXMPP /Botón Guardar los cambios y mostrar.
EC 1.4	Selecciona la opción Cancelar.								Cancela la operación.	Moodle/Agregar Actividad/ChatXMPP /Botón Cancelar.
EC 1.5	Campos vacíos	I V	V I	N/A N/A	N/A N/A	V V	N/A N/A	N/A N/A	Emite un mensaje de error sobre el/los campos sin llenar: "Debe	Moodle/Agregar Actividad/ChatXMPP /Botón Salvar los

Capítulo 3: Implementación del Módulo ChatXMPP.

		V	V	N/A	N/A	V	N/A	N/A	suministrar un valor aquí".	cambios y regresar al curso ó Botón Salvar los cambios y mostrar
EC 1.6	Campos incorrectos	V	V	N/A	N/A	I	N/A	N/A	Emite un mensaje de error sobre el campo Tiempo de Fin: "La fecha de fin debe ser mayor que la de inicio".	Moodle/Agregar Actividad/ChatXMPP /Botón Salvar los cambios y regresar al curso ó Botón Salvar los cambios y mostrar

Tabla 11 Caso de Prueba Sección Modificar Sala de Chat

Escenario	Descripción	Nombre	Descripción	Tiempo Inicio	Tiempo Fin	Modo Grupo	Visible	No ID	Respuesta del sistema	Flujo central
E C 1.1	Selecciona la opción								Permite modificar los siguientes campos:	Moodle/Agregar Actividad/ChatXMP

Capítulo 3: Implementación del Módulo ChatXMPP.

	Modificar ChatXMPP.								<ul style="list-style-type: none"> • Nombre: • Descripción • Tiempo de Inicio: • Tiempo de Fin • Modo de grupo • Visible • Número de ID <p>Y permite:</p> <ul style="list-style-type: none"> • Guardar cambios y regresar al curso • Guardar cambios y mostrar • Cancelar 	P/ Modificar ChatXMPP
EC 1.2	Introduce los datos y selecciona la opción cambios y regresar al curso.	V	V	N/A	V	N/A	N/A	N/A	<ol style="list-style-type: none"> 1. Comprueba los datos de los campos Nombre, Descripción y Tiempo de Fin. 2. Almacena los datos y crea la actividad. 	Moodle/Agregar Actividad/ChatXMP P/Botón Salvar los cambios y regresar al curso.

Capítulo 3: Implementación del Módulo ChatXMPP.

EC 1.3	Selecciona la opción Guardar cambios y mostrar								Almacena los datos y muestra la actividad creada.	Moodle/Agregar Actividad/ChatXMPP P/Botón Guardar los cambios y mostrar.
EC 1.4	Selecciona la opción Cancelar.								Cancela la operación.	Moodle/Agregar Actividad/ChatXMPP P/Botón Cancelar.
EC 1.5	Campos vacíos	I	V	N/A	N/A	V	N/A	N/A	Emite un mensaje de error sobre el/los campos sin llenar: "Debe suministrar un valor aquí".	Moodle/Agregar Actividad/ChatXMPP P/Botón Salvar los cambios y regresar al curso ó Botón Salvar los cambios y mostrar
		V	I	N/A	N/A	V	N/A	N/A		
		V	V	N/A	N/A	V	N/A	N/A		

Capítulo 3: Implementación del Módulo ChatXMPP.

EC 1.6	Campos incorrectos	V	V	N/A	N/A	I	N/A	N/A	Emite un mensaje de error sobre el campo Tiempo de Fin: “La fecha de fin debe ser mayor que la de inicio”.	Moodle/Agregar Actividad/ChatXMP P/Botón Salvar los cambios y regresar al curso ó Botón Salvar los cambios y mostrar
--------	--------------------	---	---	-----	-----	---	-----	-----	--	--

Tabla 12Caso de Prueba Sección Eliminar Sala de Chat

Escenario	Descripción	Respuesta del sistema	Flujo central
E C 1.1	Selecciona la opción Eliminar ChatXMPP.	Muestra el siguiente mensaje de confirmación: “¿Está seguro que desea borrar definitivamente el Chat?” Y permite seleccionar la opción: <ul style="list-style-type: none"> • Sí • NO 	Moodle/ChatXMPP/ Eliminar ChatXMPP
EC 1.2	Selecciona la opción Sí.	Elimina toda la información del ChatXMPP.	Moodle/ChatXMPP/ Eliminar ChatXMPP/Botón Sí

Capítulo 3: Implementación del Módulo ChatXMPP.

EC 1.3	Selecciona la opción No	Se cancela la operación.	Moodle/Agregar Actividad/ChatXMPP/Botón NO
--------	----------------------------	--------------------------	--

3.3.4 Resultados Obtenidos

“Las pruebas de software son un elemento crítico para la garantía de la calidad del software y representan una revisión final de las especificaciones, del diseño y la codificación” (Alonso, y otros, 2005).

En el gráfico que se muestra a continuación se evidencian la cantidad de no conformidades detectadas en cada una de las iteraciones por la que pasó la revisión del módulo. En la primera iteración se detectaron 3 no conformidades en la segunda una no conformidad y en la tercera no se detectó ningún error.



Figura 32 Gráfico de No Conformidades

Capítulo 3: Implementación del Módulo ChatXMPP.

Las no conformidades detectadas en la etapa de pruebas a la aplicación fueron resueltas de manera satisfactoria, contribuyendo así a la calidad del módulo desarrollado.

Conclusiones parciales

El desarrollo de este capítulo facilita la implementación del módulo ChatXMPP, debido a la organización que se adquiere por medio de los diagramas de componentes y despliegue. Además, se obtienen las relaciones de cada tabla a través del modelo de datos y se corrigen los errores existentes en la extensión por medio de los casos de pruebas aplicados al sistema.

Conclusiones

Conclusiones Generales

Con el desarrollo del siguiente trabajo de diploma se cumplieron con todas las tareas y objetivos propuestos, llegando así a las siguientes conclusiones:

- Se realizó un estudio del arte sobre la mensajería instantánea, lo cual confirmó la necesidad de un chat en la plataforma Moodle que permitiera una mejor comunicación instantánea entre los usuarios de la plataforma.
- Se efectuó un análisis de las herramientas IDE NetBeans y Visual Paradigm, los lenguajes de programación PHP, JavaScript y Ajax, el protocolo XMPP y la librería Jsjac.
- Se diseñó y programó de manera satisfactoria el módulo ChatXMPP, que facilita la comunicación instantánea entre los usuarios de la plataforma Moodle, tal como se planteó en los objetivos del trabajo.
- Se logró un producto final útil para los usuarios de Moodle, principalmente para profesores y estudiantes.
- Se comprobó que la nueva herramienta responde a los requisitos funcionales descritos, evaluando los resultados alcanzados a través de las pruebas realizadas

Recomendaciones

Aunque la herramienta que se propone supera las deficiencias existentes en cuanto a la comunicación instantánea entre los usuarios de la plataforma Moodle, aún es posible perfeccionar y ampliar las funcionalidades de la misma, para ello se recomienda:

- Comenzar la explotación del nuevo módulo dentro de la UCI, tal que pueda ser una opción a utilizar en actividades docentes y productivas.
- Integrar los resultados a la comunidad de desarrolladores de Moodle.
- Continuar perfeccionando la herramienta desarrollada, añadiendo funcionalidades tales como: incluir el envío de mensajes privados así como guardar el historial.

Bibliografía Referenciada

123flashchat. 2010. *123flashchat*. [En línea] 2010. [Citado el: 10 de Enero de 2011.] <http://www.123flashchat.com/feature.html>.

Adictosalared. 2010. Adictosalared. [En línea] 2010. [Citado el: 16 de Enero de 2011.] <http://www.adictosalared.com/holla-sistema-de-chat-gratuito-y-de-codigo-abierto-realizado-con-javascript/>.

Alonso, Fernando, Martínez, Loïc y Segovia, Fco Javier. 2005. *Introducción a la ingeniería del software*. Madrid : Delta Publicaciones, 2005.

Álvarez, Miguel Ángel. (2008). Introducción a JavaScript. Qué es JavaScript y las posibilidades que nos ofrece con respecto al HTML. [En línea] 2008. [Citado el: 15 de enero de 2011.] <http://www.desarrolloweb.com/articulos/490.php>.

Anderruthy., Jean-Noël. 2007. *SKYPE Y TELEFONÍA IP. Llama gratis por internet*. Barcelona : EDICIONES ENI , 2007. ISBN 978-2-7460-3597-3.

Berzal, Fernando, Cortijo, Francisco José y Cubero, Juan Carlos. 2005. *Desarrollo Profesional de Aplicaciones Web con ASP.NET*. Granada : Ikor Consulting, 2005. ISBN 84-609-4245-7.

Cancelo López, Pablo y Alonso Giráldez, José Miguel. 2007. *La tercera revolución: comunicación, tecnología y su nomenclatura en inglés*. La Coruña : Netbiblo, 2007. ISBN 978-84-9745-214-4.

de la Torre, Aníbal. 2006. Adelat. *Adelat*. [En línea] 2006. [Citado el: 24 de Enero de 2011.] http://www.adelat.org/media/docum/moodle/docum/23_cap01.pdf.

Eguíluz Pérez, Javier. 2007. *Introducción a Ajax*. 2007.

España Boquera, María Carmen. 2003. *Servicios avanzados de telecomunicación*. Madrid : Ediciones Días de Santos, 2003. ISBN 84-7978-607-8.

García Peñalvo, Francisco José. 2005. USAL. [En línea] 2005. [Citado el: 18 de Febrero de 2011.] http://campus.usal.es/~teoriaeducacion/rev_numero_06_2/n6_02_art_garcia_penalvo.htm.

Figuroa, Geovanni. 2011. CIDSE. [En línea] 2011. [Citado el: 13 de Febrero de 2011.] <http://www.cidse.itcr.ac.cr/revistamate/HERRAmInternet/GuionesEnJavascript/manualjs1.html>.

GSInnova. 2010. GSInnova. [En línea] 2010. [Citado el: 15 de Marzo de 2011.] <http://www.rational.com.ar/herramientas/roseenterprise.html>.

HostingGoogleProject. 2011. *XMPPHP*. [En línea] 2011. [Citado el: 6 de Enero de 2011.] <http://code.google.com/p/xmpphp/>.

Bibliografía Referenciada

IJAB. 2010. IJAB. [En línea] 2010. [Citado el: 6 de Febrero de 2011.]
<http://opensource.ijab.im>.

Jacobson, Ivan, Booch, Grady y Rumbaugh, James. 2004. *El Proceso de Unificado de Desarrollo de Software*. Madrid : Addison Wesley, 2004.

López Guzmán, Clara. 2005. Gredos Repositorio Documental. *Gredos Repositorio Documental*. [En línea] 2005. [Citado el: 15 de Enero de 2011.]
http://gredos.usal.es/jspui/bitstream/10366/56649/1/DIA_Repositoriosobjetos.pdf.pdf15
.

Moodle. 2010. moodle. [En línea] 2010. [Citado el: 8 de Febrero de 2011.]
http://docs.moodle.org/en/About_Moodle.

Musciano, Chuck y Kennedy, Bill. 2006. *HTML & XHTML: the definitive guide*. Sexta edición. s.l. : O'Reilly Media, Inc, 2006.

Pressman, Roger. 2002. *Ingeniería del software: un enfoque práctico*. Quinta edición. Madrid : McGraw-Hill, 2002. ISBN 84-481-3214-9.

PYTHON. 2005. PYTHON. [En línea] 2005. [Citado el: 24 de Enero de 2011.]

Rendón, Héctor. 2007. *El periodista digital mexicano: Hacia su definición*. 2007.

Romero Laguillo, Luis Fernando. 1998. *Publicar en Internet: guía práctica para la creación de documentos HTML*. Santander : Servicio de Publicaciones de la Universidad de Cantabria, 1998. ISBN 84-8102-179-2.

Rosenberg, Marc. 2001. *E-Learning Strategies fos Delivering Knowledge in the Digital Age*. New York : McGraw-Hill, 2001.

Sierra, Maria. 2008. [En línea] 2008. [Citado el: 9 de Enero de 2011.]
<http://personales.unican.es/ruizfr/is1/doc/lab/01/is1-p01-trans.pdf>.

SKYPE. 2011. SKYPE. [En línea] 2011. [Citado el: 13 de Enero de 2011.]
<http://www.skype.com/intl/en/features/>.

Sommerville, Ian. 2005. *Ingeniería del software*. Madrid : PEARSON EDUCATION, 2005. ISBN 84-7829-074-5.

Souto, Alejandra. 2006. *Claves Para Diseñar, Elaborar Y Aplicar Un Programa E-learning Con Exito*. Vigo : IDEASPROPIAS EDITORIAL, 2006.

St, John. 2009. Moodle. [En línea] 2009. [Citado el: 2 de Febrero de 2011.]
<http://www.moodle.org>.

Universidad de Sevilla. 2007. CFP Centro de Formación Permanente. *CFP Centro de Formación Permanente*. [En línea] 2007. [Citado el: 13 de Enero de 2011.]
<http://www.cfp.us.es/web/contenido.asp?id=3417>.

Van Lancker, Luc. 2005. *CSS en DHTML: JavaScript aplicado a hojas de estilo*. Barcelona : Ediciones ENI, 2005. ISBN 2-7460-3067-5.

Bibliografía Referenciada

Van Lancker, Luc. 2007. *XHTML 1 y CSS 1 y 2.1: los nuevos estándares de la web 2.0*. Barcelona : Ediciones ENI, 2007. ISBN 978-2-7460-3711-3.

Vasallo, Onaysi y Ramírez, Yislén. 2009. Proceso de Pruebas de Liberación al Sistema de Manejo de Datos de Ensayos. La Habana : UCI, 2009. Trabajo de Diploma.

XMPP. 2010. XMPP Standards Foundation. [En línea] 2010. [Citado el: 25 de Febrero de 2011.] <http://xmpp.org/about-xmpp/>.

Glosario de Términos

Moodle

Entorno de Aprendizaje Dinámico Orientado a Objetos y Modular; paquete de software para la creación de cursos y sitios web, que permite el diseño de cursos estructurados incluyendo recursos y actividades, y llevar un seguimiento detallado de las acciones del educando y sus avances, cuestiones necesarias para dirigir un buen proceso de aprendizaje en línea.

Chat

Chat es un término que se refiere a una comunicación escrita a través de Internet entre dos o más personas que se realiza en tiempo real.

XMPP

El protocolo XMPP (Extensible Messaging and Presence Protocol), es un protocolo abierto y extensible, con él queda establecida una plataforma para el intercambio de datos XML, que puede ser usado entre aplicaciones de Internet para mensajería instantánea, aunque originalmente fue ideado para la misma.

Mensajería instantánea

El término de Mensajería Instantánea designa a todo programa que permite dialogar en directo con un interlocutor distante mediante teclado, micrófono o teléfono.

No Conformidades

Problema detectado en un artefacto según:

- Error con respecto a lo definido en artefactos anteriores y/o en lo pactado con el cliente.
- No concordancia con Normas internacionales que deben ser cumplidas por el artefacto.
- Insatisfacción del cliente con el resultado final de un Elemento de Configuración según lo pactado con anterioridad en el proyecto.

Anexos

Tabla 13 Caso de Prueba Instalar Módulo Chat

Escenario	Descripción	Respuesta del sistema	Flujo central
EC 1.1	Selecciona la opción Notificaciones.	<ol style="list-style-type: none"> 1. Verifica si existe algún módulo por instalar. 2. Almacena la información del módulo. 3. Muestra el informe del éxito de la creación de la base de datos del módulo. 	Moodle/Administración/ Notificación.
EC 1.2	Selecciona la opción Continuar.	Incorpora el módulo en el listado de módulos reconocidos en Moodle y muestra un informe de éxito de la instalación. Finaliza el caso de uso.	Moodle/Administración/Notificación /Opción Continuar.
EC 1.3	Fracaso al almacenar información del módulo	Muestra un informe de los errores ocurridos en el proceso de creación de la base de datos.	Moodle/Administración/Notificación /Mensaje de error.

Tabla 14 Caso de Prueba Configurar Módulo Chat

Escenario	Descripción	Nombre servidor	HTTP base	Respuesta del sistema	Flujo central
EC 1.1	Selecciona la opción configurar módulo.			Solicita los datos siguientes: <ul style="list-style-type: none"> • Nombre del servidor • HTTP base 	Bloque Administración/Módulos/ChatXMPP/
EC 1.2	Datos correctos.	NA	NA	Almacena la información y muestra el mensaje "Cambios actualizados".	Bloque Administración/Módulos/ ChatXMPP/Mensaje

Tabla 15 Caso de Prueba Desinstalar Módulo

Escenario	Descripción	Respuesta del sistema	Flujo central
EC 1.1	Selecciona la opción Borrar módulo.	<p>Muestra el siguiente mensaje: “Está a punto de borrar completamente el módulo 'ChatXMPP'. Esto borrará de la base de datos todo lo asociado con este módulo. ¿Está SEGURO de que desea continuar?”</p> <p>Y permite seleccionar la opción:</p> <ul style="list-style-type: none"> • Sí • NO 	Moodle/Administración/Módulos/Actividades/Administrar Actividades/Eliminar/Mensaje
EC 1.2	Confirma la solicitud.	<ol style="list-style-type: none"> 1. Elimina toda la información del módulo de la base de datos de Moodle. 2. Muestra el siguiente mensaje “Todos los datos asociados con el módulo 'ChatXMPP' se han borrado de la base de datos. Para completar el proceso (y prevenir que el módulo se reinstale a sí mismo) debe borrar ahora el directorio /var/www/moodle/mod/Chatxmpp de su servidor”. 3. Permite seleccionar la opción Continuar y finaliza el CU. 	Moodle/Administración/Módulos/Actividades/Administrar Actividades/Eliminar/Mensaje/SÍ /Continuar.
EC 1.3	Selecciona la opción NO.	Cancela la operación y vuelve al curso.	Actividades/Administrar Actividades/Eliminar/Mensaje/ NO.

Tabla 16 Descripción de las variables Caso de Prueba Adicionar y Modificar Sala de Chat

No	Nombre de campo	Clasificación	Valor Nulo	Descripción
1	Nombre	Campo de texto	No	Campo de texto que almacena cualquier cadena de caracteres, sin tamaño límite. Acepta tanto número como letras.
2	Descripción	Campo de texto	No	Campo de texto que almacena cualquier cadena de caracteres, sin tamaño límite. Acepta tanto número como letras.
3	Tiempo Inicio	Lista desplegable	No	Lista desplegable que te permite seleccionar el día, mes, año, hora, minutos. Por defecto se muestran los datos por la configuración de la PC donde esté instalado el módulo. Ejemplo: 6/06/2011
4	Tiempo Fin	Lista desplegable	No	Lista desplegable que te permite seleccionar el día, mes, año, hora, minutos. Por defecto se muestran los datos por la configuración de la PC donde esté instalado el módulo. Siempre mayor que el Tiempo inicio. Ejemplo: 17/06/2011
5	Modo Grupo	Lista desplegable	No	Lista desplegable que te permite seleccionar a qué grupo estará visible el Chat. Por defecto sale uno de las tres opciones: Grupos separados, No hay grupos y Grupos visibles.

Anexos

6	Visible	Lista desplegable	No	Lista desplegable que indica si la actividad estará visible o no para todos los usuarios del curso. Por defecto sale uno de las tres opciones: Mostrar u Ocultar.
7	No ID	Campo de texto	Sí	Campo de texto que almacena cualquier cadena de caracteres, siempre y cuando no se repita, es decir, número único.

Tabla 17 Caso de Prueba Autenticar usuario

Escenario	Descripción	Usuario	Contraseña	Respuesta del sistema	Flujo central
EC 1.1	Llena el campo de la contraseña correctamente.			Muestra la opción para realizar la autenticación del usuario, donde debe recoger los siguientes datos: <ul style="list-style-type: none"> • Usuario. • Contraseña. 	Moodle/Curso/Nombre Sala de chat/ Link "Click aquí para entrar al ChatXMPP" /Autenticarse
EC 1.2	Introduce los datos y selecciona la opción de entrar.	N/A	V	1. Valida los datos. 2. Actualiza en la base de datos los datos del usuario autenticado.	Moodle/Curso/Nombre Sala de chat/ Link "Click aquí para entrar al ChatXMPP" /Autenticarse
EC 1.3	Fallo en la autenticación	N/A	I	Muestra un mensaje de error indicando falló la autenticación.	Moodle/Curso/Nombre Sala de chat/ Link "Click aquí para entrar al ChatXMPP" /Autenticarse/Botón OK

Tabla 18 Descripción de las variable Caso de Prueba Autenticar Usuario

No	Nombre de campo	Clasificación	Valor Nulo	Descripción
1	Usuario	Campo de texto	No	Campo de texto, tiene por defecto los valores que puede almacenar. Ejemplo: rpgarcia.
2	Contraseña	Campo de texto	No	Campo de texto que acepta cualquier cadena de caracteres que acepta tanto número como letras. Ejemplo: Contrasen@123*

Tabla 19 Caso de Prueba Entrar Sala de Chat

Escenario	Descripción	Respuesta del sistema	Flujo central
EC 1.1	Selecciona la opción Entrar a la sala de chat.	Muestra la ventana de la sala de Chat, los usuarios que se encuentran conectados y los mensajes enviados.	Curso/Agregar Sala de chat/ Link "Click aquí para entrar en el ChatXMPP ahora"/Entrar Sala de Chat.
EC 1.2	Selecciona la opción Entrar sala de Chat fuera de fecha.	El sistema muestra un mensaje de error "No puede seleccionar esta Sala de Chat porque está fuera de fecha". Y muestra el botón Continuar.	Curso/Agregar Sala de chat/ Link "Click aquí para entrar en el ChatXMPP ahora"/Entrar Sala de Chat.

Tabla 20 Caso de Prueba Mostrar Listado de Usuarios Conectados

Escenario	Descripción	Respuesta del sistema	Flujo central
EC 1.1	Mostrar usuarios conectados en la sala de Chat.	Muestra los usuarios conectados en la Sala de Chat.	Curso/Nombre sala de chat/ Link "Click para entrar al ChatXMPP ahora"/Entrar Sala de Chat.