



**HERRAMIENTA PARA PROVEER AYUDAS ADAPTATIVAS EN AMBIENTES E-LEARNING
UTILIZANDO AGENTES SOFTWARE**

**LUZ ADRIANA PEÑARANDA GUEVARA
0767324**

**FACULTAD DE INGENIERÍA
ESCUELA DE INGENIERÍA DE SISTEMAS Y COMPUTACIÓN
PROGRAMA ACADÉMICO DE INGENIERÍA DE SISTEMAS
TULUÁ, NOVIEMBRE 01 DE 2013**



**HERRAMIENTA PARA PROVEER AYUDAS ADAPTATIVAS EN AMBIENTES E-LEARNING
UTILIZANDO AGENTES SOFTWARE**

LUZ ADRIANA PEÑARANDA GUEVARA
0767324
luzana318@gmail.com

JAVIER M. REYES VERA
MAGÍSTER (C) EN INGENIERÍA DE SISTEMAS.
DOCENTE DEPARTAMENTO DE DISEÑO.
UNIVERSIDAD DEL VALLE
javier.reyes@correounivalle.edu.co

FACULTAD DE INGENIERÍA
ESCUELA DE INGENIERÍA DE SISTEMAS Y COMPUTACIÓN
PROGRAMA ACADÉMICO DE INGENIERÍA DE SISTEMAS
TULUÁ, NOVIEMBRE 01 DE 2013

Notas de aceptación.

Firma Jurado.

Firma Jurado.

Firma Director de
Proyecto de Grado.

Firma Coordinador de
Ingeniería de Sistemas – Tuluá.

AGRADECIMIENTOS

Deseo agradecerle primero a Dios por haberme dado esta oportunidad de hacer realidad mi sueño de estudiar Ingeniería en Sistemas, y por haberme dado a mis padres Marco Fidel Peñaranda C. y Sul Lucrecia Guevara G. quienes han sido mi apoyo y mi guía en cada momento.

A mi hermano Alejandro Peñaranda G, a mis tíos Segundo A. Guevara G, Hernando Guevara G, a mi prima Melissa Guevara, quienes siempre han creído en mí y siempre darme su apoyo.

A mi director Javier M. Reyes V. Por su valiosa colaboración en el desarrollo del proyecto. Por su confianza, acompañamiento, amabilidad y apoyo a lo largo de este tiempo.

A la profesora Liliana Machuca por sus aportes, acompañamiento, y apoyo en el proceso. El aporte que realizó al proyecto ha sido importante. Y a todas las personas del proyecto Lingweb por su acompañamiento y apoyo.

Tabla de Contenido

RESUMEN.....	9
1 INTRODUCCIÓN.....	10
2 PLANTEAMIENTO Y FORMULACIÓN DEL PROBLEMA.....	12
3 JUSTIFICACIÓN.....	13
4 OBJETIVOS.....	13
4.1 OBJETIVO GENERAL.....	13
4.2 OBJETIVOS ESPECÍFICOS.....	13
5 ALCANCE DE LA PROPUESTA.....	13
5.1 RESULTADOS ESPERADOS.....	14
6 MARCO DE REFERENCIA.....	15
6.1 GLOSARIO.....	15
6.2 MARCO TEÓRICO.....	15
6.2.1 PERSONALIZACIÓN.....	15
6.2.2 AGENTES SOFTWARE.....	15
6.2.3 SISTEMAS MULTIAGENTES.....	17
6.2.4 SISTEMAS DE AYUDA.....	17
6.2.5 ARQUITECTURA LINGWEB.....	18
6.2.5.1 Vista conceptual.....	18
6.2.5.1 Vista lógica.....	19
6.3 ESTADO DEL ARTE.....	20
7 ESPECIFICACIÓN DEL PROTOTIPO DE LA HERRAMIENTA PARA PROVEER AYUDAS ADAPTATIVAS EN AMBIENTES E-LEARNING.....	23
7.1 DESCRIPCIÓN Y SELECCIÓN DE LAS AYUDAS QUE OFRECERÁ EL PROTOTIPO.....	24
7.2 DESCRIPCIÓN ENFOQUE PARA LA ASIGNACIÓN Y CREACIÓN DE LAS AYUDAS DEL PROTOTIPO.....	26
7.3 DESCRIPCIÓN Y SELECCIÓN DEL METODO DE MONITOREO DE USUARIO DEL PROTOTIPO.....	27
7.4 DESCRIPCIÓN DE LOS ALGORITMOS DE LOS MÉTODO DE MONITOREO DE USUARIO Y DE ASIGNACIÓN DE AYUDAS DEL PROTOTIPO.....	28
7.4.1 MÉTODO DE MONITOREO DE USUARIOS.....	29
7.4.1.1 Algoritmo para el Monitoreo de los eventos del mouse sobre los iconos, enlaces y botones.....	30
7.4.1.1.1 Cinco casos de los eventos de los usuarios.....	30
7.4.1.1.1.1 clic o doble clic en los iconos o enlaces.....	30
7.4.1.1.1.2 clic derecho en los enlaces.....	32
7.4.1.1.1.3 clic en los campo de texto del formulario de crear curso.....	32
7.4.1.1.1.4 clic en los botones del formulario en especial el icono de ayuda de los campos de texto..	33
7.4.1.1.1.5 en caso de que no se reciban eventos del usuario.....	33
7.4.2 MÉTODO DE ASIGNACIÓN DE AYUDAS.....	34
7.4.2.1 Algoritmo para para la asignación de las ayudas.....	34
7.4.2.1.1 Ayuda para el caso clic derecho en los enlaces.....	35
7.4.2.1.2 Ayuda para el caso clic en los botones del formulario en especial el icono de ayuda de los campos de texto.....	35
7.4.2.1.3 Ayuda para el caso de que no se reciban eventos del usuario.....	36
7.5 MODELO DE AGENTES PARA PROVEER AYUDAS ADAPTATIVAS.....	37
7.5.1 MODELO DE OBJETIVOS.....	37
7.5.2 MODELO DE AGENTES.....	38
7.5.3 MODELO DE ORGANIZACIÓN.....	39

7.5.4 MODELO DE ENTORNO.....	40
7.5.5 MODELO DE SERVICIOS.....	41
8 DESARROLLO PROTOTIPO.....	41
8.1 METODOLOGÍA DE DESARROLLO.....	41
8.2 TECNOLOGÍAS UTILIZADAS.....	43
8.4 ANÁLISIS DE REQUERIMIENTOS.....	45
8.5 MODELO GENERAL DE INTEGRACION ENTRE SERVLETS Y JADE DEL SISTEMA MULTIAGENTE DE LINGWEB.....	48
8.6 DISEÑO.....	48
8.6.1 DISEÑO COMPUTACIONAL.....	48
8.6.1.1 MODELO DE NAVEGACIÓN.....	48
8.6.1.2 Diagrama de clases.....	49
8.6.1.3 Modelo de persistencia.....	50
9 PRUEBAS.....	56
9.1 PPRUEBAS UNITARIAS.....	56
9.2 PRUEBAS DE INTEGRACIÓN.....	58
9.2.1 DETALLE DE LA PRUEBA.....	56
10 CONCLUSIÓN.....	59
10. 1 DE LA HERRAMIENTA.....	59
10.2 DE LAS METODOLOGÍAS USADAS.....	59
10.3 PERSONALES.....	59
11 TRABAJO FUTURO.....	59
11. 1 Herramienta para proveer ayudas adaptativas.....	59
11. 2 Plataforma Lingweb.....	59
12 REFERENCIAS.....	60

Listas de figuras

Figura 1. Agente de interfaz.....	16
Figura 2. Comportamiento agente.....	17
Figura 3. Estructura Conceptual del Sistema.....	18
Figura 4. Vista conceptual.....	18
Figura 5. Estructura general por capas.....	19
Figura 6. Estructura del Paquete Adaptatividad.....	20
Figura 7. Guía y Asistencia para Entornos e-learning del Prototipo.....	24
Figura 8. Apariencia de Lingweb.....	29
Figura 9. Icono, enlaces de las herramientas de Lingweb que monitorea el prototipo.....	30
Figura 10. Botones del formulario de cursos de Lingweb que monitorea el prototipo.....	30
Figura 11. Lista de cursos de Lingweb.....	32
Figura 12. Formulario crear Curso Lingweb.....	32
Figura 13. Opción de crear cursos de Lingweb.....	35
Figura 14. Ayuda sabias qué?, creada por el agente ayuda.....	35
Figura 15. Ayuda comando qué es esto?, creada por el agente ayuda.....	35
Figura 16. Ayuda contextual ayudante creada por el agente ayuda.....	36
Figura 17. Ayuda de referencia creada por el agente ayuda.....	36
Figura 18. Modelo de Objetivo General.....	37
Figura 19. Modelo de Sub-objetivo Monitorear Usuarios.....	37
Figura 20. Modelo de Sub-objetivo Mostrar Ayuda.....	37
Figura 21. Detalle del Modelo de Agentes Monitor y Ayuda.....	38
Figura 22. Modelo de Organización.....	40
Figura 23. Modelo de Entorno.....	40
Figura 24. Modelo de Servicios: comunicar eventos del usuario.....	41
Figura 25. Modelo de Servicios: pedir ayuda.....	41
Figura 26. Modelo de Servicios: mostrar ayuda.....	41
Figura 27. Usuarios Lingweb y usuario del prototipo de ayudas.....	45
Figura 28. Modelo general de integración Servlets y Jade.....	47
Figura 29. Arquitectura LingWeb en tres capas.....	48
Figura 30. Modelo de navegación del monitoreo de usuarios.....	49
Figura 31. Diagrama de clases del monitoreo de usuarios.....	50
Figura 32. Modelo persistencia del monitoreo de usuarios.....	51
Figura 33. Modelo persistencia del monitoreo de usuarios.....	58

Índice de Tablas

Tabla 1: Relación de resultados.....	14
Tabla 2. Tipos de ayudas del modelo.....	26
Tabla 3. Eventos sobre los iconos y enlaces.....	31
Tabla 4. Eventos sobre los campos de texto.....	33
Tabla 5. Responsabilidades de los agentes.....	39
Tabla 6. Fases del proceso de desarrollo en meses.....	42
Tabla 7. Funciones de las aplicaciones distribuidas por perfil de usuario.....	43
Tabla 8. Funciones de las aplicaciones distribuidas por perfil de usuario.....	46
Tabla 9. Estructura de los requerimientos en el SRS.....	46
Tabla 10. Requerimiento creación ejercicio de apareamiento.....	47
Tabla 11. Elementos pruebas unitarias.....	57
Tabla 12. Pruebas integración.....	59

Lista Anexos

ANEXO I - Marco de Referencia

ANEXO II - Documento SRS - Software Requirements Specification

ANEXO III - Modelos de Agentes

ANEXO IV - Documento Método de Monitoreo y Asignación de Ayudas

ANEXO V - Integración Jade y Servlets

ANEXO VI - Modelo de persistencia y diagramas de clases

ANEXO VII - Manual de Instalación Lingweb

ANEXO VIII - Manual de Usuario Lingweb

ANEXO IX - Instrucciones para usar prototipo

RESUMEN

Este trabajo propone el desarrollo de una herramienta para proveer ayudas adaptativas en ambientes e-learning, centrado en la personalización de las ayudas empleando la tecnología de agente software. La finalidad de utilizar agentes software es proveer inteligencia, es decir, adaptatividad¹ a las ayudas en ambientes e-learning, teniendo en cuenta las necesidades de los usuarios y su desenvolvimiento en la plataforma, para lo cual se creó un prototipo y un método que permite la asignación y seguimiento de las ayudas a los usuarios de ambientes e-learning. Como caso de estudio se trabajó con la plataforma Lingweb “Ambiente Virtual para el aprendizaje de Idiomas”. Lingweb fue desarrollado entre diciembre de 2007 y junio de 2010 en la modalidad de cofinanciación entre la Universidad del Valle y Colciencias².

Palabras Clave: ayudas adaptativas, personalización, agentes inteligentes, etc.

¹Adaptatividad :Facultad del sistema para realizar automáticamente cambios en el entorno de trabajo del usuario, según sea su perfil.

²Para mayor información visitar <http://lingweb.univalle.edu.co>

1. INTRODUCCIÓN

En la actualidad los sistemas e-learning ya sean de libre distribución o licenciados, ofrecen el mismo estilo de ayudas para todos los usuarios. Estas ayudas normalmente son tutoriales, manuales de usuario, ayuda en línea, y no tienen en cuenta las individualidades de los usuarios, y pueden llegar incluso a dejar a los usuarios sin la información necesaria para el uso de la aplicación. Debido a lo anterior, cada vez resulta más importante que los usuarios de los ambientes virtuales puedan disponer de ayudas que faciliten la utilización y comprensión de dichas plataformas.

Las ayudas de las plataformas virtuales, en su mayoría están pensadas de modo general, es decir, el contenido de estas ayudas es igual para todos los usuarios y no consideran sus particularidades, por esta razón, contar con ayudas personalizadas; puede ser una estrategia para mejorar la flexibilidad de las aplicaciones de estas plataformas, donde se tenga en cuenta los usuarios de acuerdo con sus modelos mentales individuales.

El propósito de este trabajo consistió en desarrollar un sistema de ayuda que permita ofrecer a los usuarios características de adaptatividad³, usando una metodología de desarrollo denominada RUP ágil, que consiste en combinar los artefactos generados en la tecnología RUP con las iteraciones utilizadas en la tecnología XP o más conocida como programación extrema.

Existen muchas técnicas para ofrecer personalización, como por ejemplo el método de personalización de la información, el método de presentación adaptativa, los Agentes Software, entre otros. El método de personalización de la información se puede definir como la adaptación dinámica e inteligente de la información genérica sobre la base de las características del usuario más destacados como la demografía de los usuarios, los conocimientos, las habilidades, la personalidad, los intereses, los gustos, las preferencias, los fines, las necesidades, las metas, los planes, las actitudes y de comportamiento, para efectuar una experiencia de mediación de información personalizada para el usuario [Uchyigit 2009]. El método de presentación adaptativa consiste en adaptar el contenido de una página visitada por un usuario en particular con los conocimientos actuales, los objetivos, y otras características del usuario [Brusilovsky 1996]. Sin embargo los agentes software se proponen para alcanzar este fin como una posibilidad importante ya que permiten la solución de problemas en los que hay múltiples métodos de solución y/o múltiples entidades capaces de trabajar conjuntamente para solucionarlos.

La tecnología de Agentes de Software ha sido un campo importante de investigación durante muchos años y una alternativa bastante atractiva para aplicaciones web; de la misma manera que las denominadas RIA (acrónimo de Rich Internet Applications), conocidas porque hacen posible que una aplicación Web se comporten como una aplicación de escritorio.

El propósito de éste proyecto, es la creación del prototipo de la herramienta para proveer ayudas adaptativas en ambientes e-learning. Como caso de estudio se trabajó con la plataforma Lingweb, el cual es un ambiente virtual basado en agentes de software para la enseñanza y aprendizaje de idiomas, este proyecto fue desarrollado entre diciembre de 2007 y junio de 2010 en la modalidad de cofinanciación entre la Universidad del Valle y Colciencias [Rodríguez y Machuca 2011]. Ya que para la creación de prototipo se necesita que un sistema e-learning ya existente al cual adaptarlo. Lingweb cuenta con tres tipos de usuarios administrador, profesor y estudiante el prototipo se desarrollara el lado del usuario profesor.

El aporte de este proyecto haciendo uso de la tecnología de agentes software, es exponer la creación del prototipo de ayudas adaptativas a partir del modelo de agentes software que permitirán la asignación de las

³Adaptatividad: Facultad del sistema para realizar automáticamente cambios en el entorno de trabajo del usuario, según sea su perfil.

ayudas y el monitoreo de los usuarios, tomando en cuenta las características de los usuarios, es decir su perfil, su interacción con el aplicación, para ofrecer la ayuda o las ayudas necesarias y los diferentes conocimientos que fueron necesarios para su construcción. En este proyecto también se expone los métodos para la asignación de las ayudas y el monitoreo de los usuarios de ambientes e-learning.

2. PLANTEAMIENTO Y FORMULACIÓN DEL PROBLEMA

Las principales funcionalidades que poseen la mayoría de las plataformas de e-learning son: la gestión del curso, el control de progreso, la creación de índices, los libros de notas, las listas de distribución, los espacios para anuncios, los foros de discusión, el chat, la agenda, las auto-evaluaciones, el correo electrónico interno, la auto-matrícula, la autenticación, la capacidad de elegir un idioma, y la capacidad de cambiar la apariencia, entre otros. Esto demuestra que la gran mayoría de sus funcionalidades se centran en la apariencia de la aplicación y en los contenidos. [Boneu 2007]

Además los sistemas e-learning brindan el mismo estilo de ayuda a todos sus usuarios, las cuales, normalmente incluyen tutoriales, manuales de usuario, ayuda en línea, por teléfono o correo electrónico. Y que pueden ser insuficientes para los usuarios básicos que son los que más hacen uso de estas ayudas.

Con la creciente complejidad de las aplicaciones web, la necesidad de desarrollar funciones de apoyo fáciles de usar también crece, por esta razón surgen las ayudas adaptativas, que ofrecen la cantidad adecuada de información en el momento oportuno mediante la adaptación de los contenidos del sistema de ayuda tanto para el conocimiento como los planes actuales y los objetivos de cada usuario. [Iglezaki 2004]

La personalización es ahora reconocida como una característica importante para las aplicaciones Web. Los resultados del estudio y análisis de mercado han concluido que los usuarios prefieren una experiencia web personalizada; y se observa con interés la aparición de características de personalización básicas que ofrecen los sitios web populares; como My Yahoo, My MSN y My Google. Demuestran la importancia de la personalización en aplicaciones web. [Raza 2009]

Es debido a ésto, que surge la necesidad de brindarle a los usuarios de estos sistemas e-learning ayudas personalizadas que ofrezcan la información necesaria y precisa para el uso de estos sistemas.

La solución que se plantea para este problema es brindar personalización a partir del uso de agentes software ya que posibilitan la solución de problemas en los que hay múltiples métodos de solución y/o múltiples entidades capaces de trabajar conjuntamente para solucionarlos. También se proponen dos métodos uno para la asignación de las ayudas y el otro para monitoreo de los usuarios y de las ayudas utilizadas por cada usuario, esto con el fin de asegurar que las ayudas sean las correctas y que brinden al usuario la cantidad de información necesaria para llevar a cabo sus tareas en la plataforma.

Lingweb es un Ambiente virtual para el aprendizaje de los idiomas, basado en una arquitectura multi-agente para la enseñanza y aprendizaje de idiomas (se explica por completo en [Aricapa y Puentes 2009], [Machuca 2009], [Machuca y Rodríguez 2011], [Rodríguez y Machuca 2009]). Lingweb como la mayoría de los plataformas e-learning cuenta con el mismo estilo de ayuda para todos los usuarios, por lo cual se tomó como caso de estudio para éste proyecto, por lo que se creó el prototipo dentro de la plataforma Lingweb, acoplándose a todas las especificaciones de programación del proyecto Lingweb.

El prototipo se realizó solo para el usuario profesor, el cual cuenta con un segundo perfil que representa el nivel de conocimiento del usuario sobre la plataforma y que son usuario básico, usuario intermedio y usuario avanzado.

3. JUSTIFICACIÓN

A través del desarrollo de este trabajo de investigación se contribuye a:

- Proponer el uso de Agentes Software para el desarrollo de ayudas adaptativas en ambientes e-learning.
- Desarrollar un prototipo funcional que permita analizar la pertinencia del uso de Agentes Software en el campo de la aplicación.
- Dotar de inteligencia y adaptatividad a sistemas de ayudas para las plataformas e-learning centrados en la enseñanza.
- Monitorear los movimientos de los usuarios de un sistema de e-learning para proveerles asistencia a los usuarios.

4. OBJETIVOS

4.1 OBJETIVO GENERAL

Implementar un sistema de ayudas adaptativas basadas en agentes software que permita apoyar procesos de interacción de los usuarios con la funcionalidad de las herramientas en un ambiente e-learning.

4.2 OBJETIVOS ESPECÍFICOS

- Diseñar un modelo de ayudas adaptativas para ambientes e-learning, para brindarle soporte en el uso de la aplicación a los usuarios.
- Generar un método para la asignación y monitoreo de las ayudas a los usuarios de ambientes e-learning.
- Construir un prototipo que permita asignar y monitorear las ayudas adaptativas, dependiendo del perfil y la interacción del usuario con la aplicación.

5. ALCANCE DE LA PROPUESTA

El software diseñado y desarrollado, es utilizado en la plataforma Lingweb, plataforma de enseñanza de idiomas de la Universidad del Valle, con el fin de mejorar el sistema de ayuda de esta plataforma para la comunidad universitaria.

Estas aplicaciones son acopladas a la plataforma, afectando los módulos profesor, cursos y actividades.

Para cumplir lo previamente estipulado, este proyecto tendrá en cuenta:

- El prototipo a construido se limitó solo al modelo de agentes propuesto.
- La magnitud del sistema resultante y las limitaciones de tiempo impuestas para la ejecución del proyecto.
- La organización estructural y lógica en la que está compuesto el proyecto Lingweb.
- La integración de las aplicaciones realizadas con la plataforma Lingweb. Este procedimiento se realizó tanto en un entorno de pruebas, como de creación del prototipo.
- Se realizaron pruebas unitarias y de integración sobre las aplicaciones realizadas.
- Se excluyen algunas pruebas como aceptación, piloto o de campo sobre el prototipo a desarrollar, debido a que sólo constituye una parte del sistema.
- La creación del segundo perfil para los usuarios, se asume que los perfiles ya están definidos, que son usuario básico, intermedio y avanzado.

- El prototipo se realizó solo para el usuario profesor, el cual cuenta con un segundo perfil que representa el nivel de conocimiento del usuario sobre la plataforma y que son usuario básico, usuario intermedio y usuario avanzado.

El presente trabajo no se tuvo en cuenta:

- Las ayudas pedagógicas, pues estas se salen del contexto del trabajo.
- No se tendrán en cuenta usuarios en situación de discapacidad.

5.1 RESULTADOS ESPERADOS

Objetivos Específicos	Producto(s) Esperados	Sección del Documento
Diseñar un modelo de ayudas adaptativas para ambientes e-learning, para brindarle soporte en el uso de la aplicación a los usuarios.	Documento de definición del modelo que sirva como referencia para futuros desarrollos.	<ul style="list-style-type: none"> • Sección 6.2.5 Arquitectura Lingweb. • Anexo III (Documento Modelo de Agentes y casos de uso)
Generar un método para la asignación y monitoreo de las ayudas a los usuarios de ambientes e-learning.	Documentación del método para asignar y monitoreo de los usuarios y las ayudas.	<ul style="list-style-type: none"> • Anexo III (Documento Modelo de Agentes y casos de uso) • Anexo IV (Documento Métodos de monitoreo y asignación de ayudas) • Sección 9 Pruebas
Construir un prototipo que permita asignar y monitorear las ayudas adaptativas, dependiendo del perfil y la interacción del usuario con la aplicación.	<ul style="list-style-type: none"> - Prototipo. - Implementación y Pruebas - Código fuente y ejecutable. - Interfaces de las Ayudas. - Documentación. 	<ul style="list-style-type: none"> • Sección 6.2.5 Arquitectura Lingweb. • Sección 7.5 Modelo multiagente para proveer ayudas adaptativas. • Sección 9 Pruebas

Tabla 1: Relación de resultados

Referencia de Artefactos Generados:

- Etapa de análisis: Documento de Especificación de requerimientos
- Etapa de diseño: El modelo de navegación, modelo de persistencia, el diagrama de clases.
- Etapa de implementación: El código fuente, el ejecutable y el manual de usuario.
- Etapa de pruebas: Resultados de ejecución de pruebas funcionales.

El resultado final será el desarrollo de la herramienta para proveer ayudas adaptativas en ambientes e-learning utilizando agentes software.

6. MARCO DE REFERENCIA

6.1 GLOSARIO

AUTONOMÍA: Capacidad que tiene un agente de actuar sin intervención humana directa o de otros agentes. [Molina 2004]

MAS – SMA: MultiAgent System – Sistemas Multiagente

SISTEMAS MULTIAGENTE: estos sistemas están basados en unidades (agentes), en los que se fusionan las capacidades individuales de cada uno y se identifican características de comunicación, coordinación y organización, que son fundamentales para este tipo de sistemas. [Jimenez 2000]

REACTIVIDAD: un agente está inmerso en un determinado entorno (hábitat), del que percibe estímulos y ante los que debe reaccionar en un tiempo preestablecido. [Molina 2004]

SISTEMA ADAPTATIVO: es “aquel que, basado en el conocimiento, altera automáticamente aspectos de funcionalidad e interacción para lograr acomodar las distintas preferencias y requerimientos de sus distintos usuarios.” [Deagostini & Cormenzana 2007]

SISTEMAS ELEARNING: Sistemas de administración de aprendizaje virtual.

6.2 MARCO TEÓRICO

A continuación se listan algunas definiciones, que han sido tomados como base inicial para la estructuración teórica de la propuesta, intentando buscar los aspectos relacionados con los temas a tratar en el proyecto como son: la personalización, los agentes inteligentes, los Sistemas Multiagentes y sistemas de ayudas y el sistema Lingweb donde se incluyó el prototipo.

6.2.1 PERSONALIZACIÓN

Personalización: La personalización puede ser definida como “el proceso de habilitación de un sistema para adaptar la información a las necesidades de sus usuarios y sus preferencias”. [Uchyigit 2009]

Tipos de personalización:

- **Personalización de la presentación:** Está relacionado con la presentación de la interfaz para el usuario, incluyendo los colores, tipos de letra y la posición en la ventana. [Velásquez Y Palade, 2008]
- **Personalización del contenido:** Esta es la parte más compleja, donde la información puede ser adaptada a las necesidades particulares de los usuarios. [Velásquez Y Palade, 2008]
- **Personalización de información:** la personalización Inteligente de la Información se puede definir como la adaptación dinámica e inteligente de la información genérica sobre la base de las características del usuario más destacados como la demografía de los usuarios, los conocimientos, las habilidades, la personalidad, los intereses, los gustos, las preferencias, los fines, las necesidades, las metas, los planes, las actitudes y de comportamiento cualquier otro criterio a otros específicos del usuario efectuar una experiencia de mediación de información personalizada para el usuario. [Uchyigit 2009]

6.2.2 AGENTES SOFTWARE

Agente: “Un agente es un sistema informático que se encuentra en un entorno, y que es capaz de actuar con autonomía en este ámbito con el fin de cumplir con sus objetivos de diseño. [Padgham & Winikoff 2004]

Agentes Inteligentes

Agente Inteligente: Son entidades de software que llevan a cabo un conjunto de operaciones sobre el comportamiento del usuario o cualquier otro programa con un grado de independencia o autonomía. [Peña 2004]

Un agente inteligente es una pieza de software que es:

- Situado: existe en un ambiente
- Autónomos: independiente, no controlada externamente
- Reactiva: responde (en el momento oportuno) a los cambios en su medio ambiente
- Proactivo: persistentemente persigue objetivos
- Flexible: tiene múltiples formas de lograr las metas
- Robusto: se recupera de un fallo
- Social: interactúa con otros agentes [Padgham & Winikoff 2004]

Es posible clasificar los agentes de software, de acuerdo a sus características, en reactivos e inteligentes; y de acuerdo a su utilidad en agentes de interfaz, de monitoreo, de recuperación de información, recomendación y negociación o comercialización [Jimenez 2000].

Los Agentes de Interfaz son capaces de proveer asistencia al usuario de alguna aplicación; están encargados de monitorear las acciones del usuario y desarrollar modelos sobre sus habilidades para ayudarlo cuando tenga problemas. Estos agentes, son programas que utilizan técnicas de Inteligencia Artificial y cuya implementación puede estar enfocada de diferentes maneras, algunas de ellas son: La aplicación como un agente de interfaz; la aplicación basada en el conocimiento y la aplicación basada en aprendizaje [Guerra 1997].

Ejemplo de un agente de interfaz se refleja en la figura 1.

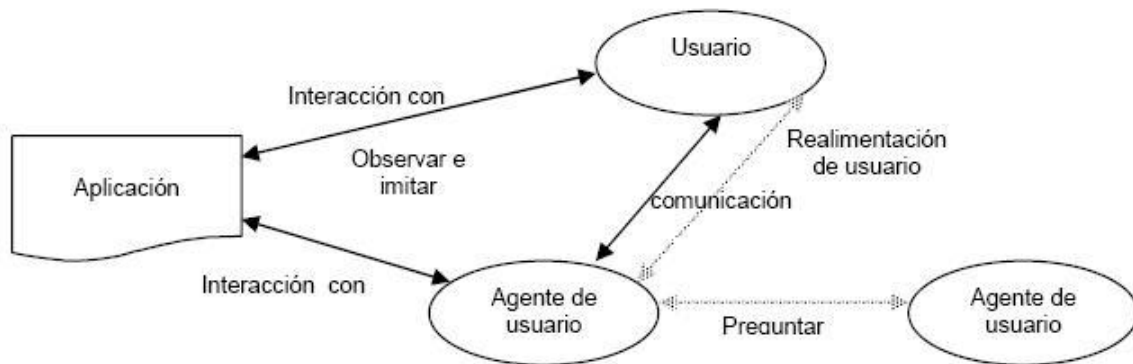


Figura 1. Agente de interfaz [Santacruz V. Liliana P. 2008]

El comportamiento de un agente se refleja en la figura 2.

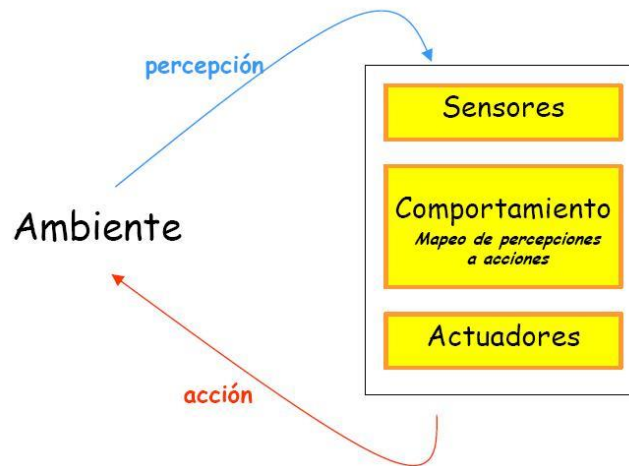


Figura 2. Comportamiento agente [Bedoya 2003]

6.2.3 SISTEMAS MULTIAGENTES

Dentro de los agentes de software, se destacan los sistemas MultiAgentes (MAS), que son un conjunto de agentes autónomos, generalmente heterogéneos y potencialmente independientes, que trabajan en común para resolver un problema. Los agentes se caracterizan por ser capaces de tomar la iniciativa, ser capaces de compartir conocimiento, ser capaces de cooperar y negociar, ser capaces de comprometerse con metas comunes. [Mas 2005], [Peña 2004].

6.2.4 SISTEMAS DE AYUDA

En la actualidad existen diversos sistemas y herramientas para proveer ayudas, cuyo objetivo principal es ayudar a los usuarios de diferentes aplicaciones tanto web, como de escritorio en el desenvolvimiento de las mismas. Estos sistemas son conocidos como sistemas de ayuda.

Los sistemas de ayuda para aplicaciones informáticas son programas que asisten al usuario en su utilización. Por esta razón su objetivo principal es el de proporcionar respuestas a los problemas que se presentan en el uso de la aplicación [Iglesakis 2004].

Existen diversas maneras de ofrecer asistencia o ayuda a los usuarios, unas de ellas son los sistemas de ayuda inteligente [Breuker 1990], [Winkels 1992] y los sistemas de ayuda adaptativos [Iglesakis 2004]. Los sistemas de ayuda inteligente (SAI) o asistentes inteligentes, son una propuesta para mejorar el aprendizaje y el rendimiento en la utilización de entornos informáticos complejos. Habitualmente la ayuda tiene como único fin la mejora del rendimiento del usuario, por ejemplo, en el Sistema de Ayuda Inteligente Aran o SAI Aran, se plantea que la asistencia debe tener también un propósito educativo. Si el usuario mejora su conocimiento sobre la aplicación su necesidad de ayuda decrecerá con el tiempo. Un punto clave de los SAI es el énfasis que se hace en la adaptación de la ayuda proporcionada a las necesidades y los conocimientos específicos de cada usuario [Fernández 2001].

El modelo de usuario del SAI Aran, es un modelo individual y dinámico basado en estereotipos. El modelado mediante estereotipos se basa en la identificación de grupos de usuarios que tengan ciertas características homogéneas respecto a la aplicación. Para cada estereotipo se identifican un pequeño número de características clave que permiten al sistema identificar si un usuario se puede clasificar o no dentro de un determinado grupo. De esta forma a un usuario concreto se le pueden aplicar uno o varios estereotipos. [Kearsley 1988], [Fernández 2001].

6.2.5 ARQUITECTURA LINGWEB

Lingweb es un Ambiente virtual basado en una arquitectura multi-agente para la enseñanza y aprendizaje de idiomas [Rodríguez y Machuca 2009]. Lingweb como la mayoría de las plataformas e-learning cuenta con el mismo estilo de ayuda para todos los usuarios, por lo cual se tomó como caso de estudio para éste proyecto; dado que presentaba la problemática a la cual se da una posible solución en este proyecto. El prototipo se creó dentro de la plataforma Lingweb, acoplándose a todas las especificaciones de programación del proyecto Lingweb, es decir, el modelo de agentes propuesto se integró al sistema multiagentes que tiene Lingweb, junto con todas las clases que conforman el prototipo.

Lingweb es una plataforma que brinda un entorno de trabajo para profesores y estudiantes para la enseñanza y aprendizaje de idiomas. Integra funcionalidades de interactividad como el chat, el foro, el diario entre otros, también cuenta con funcionalidades de diseño de tareas y actividades, de comunicación y socialización, de evaluación y retroalimentación; estas funcionalidades son características propias de los sistemas e-learning. Lingweb además de ofrecer estas funcionalidades básicas, también cuenta con herramientas para el procesamiento lingüístico como el editor avanzado y las plantillas de escritura con ayudante pedagógico, que otras plataformas no tienen y que la definen como un ambiente propicio para el desarrollo de habilidades en el estudio de un segundo idioma. [Machuca y Rodríguez 2011]

Lingweb es una aplicación que se basa en una arquitectura de tres capas que permite la mantenibilidad y un mejor entendimiento del código; además, es extensible y portable entre plataformas (la arquitectura y demás trabajos relacionados con Lingweb se explica por completo en [Aricapa y Puentes 2009], [Machuca 2009], [Machuca y Rodríguez 2011], [Rodríguez y Machuca 2009]).

A continuación se describe el conjunto de vistas que conforman la arquitectura de la plataforma Lingweb, haciendo énfasis sólo en las partes de la plataforma en las que el modelo de agentes intervendrá, y teniendo en cuenta que el modelo se acoge por completo a la arquitectura ya establecida de Lingweb.

6.2.5.1 Vista conceptual

Figura 3 se identifican las secciones o módulos en los que se ha organizado la plataforma Lingweb. Cada sección se ha agrupado teniendo en cuenta los patrones de diseño alta cohesión y bajo acoplamiento, esto ha permitido que se generen las áreas que se visualizan en la figura 3, de acuerdo a las funcionalidades comunes presentadas por ellas [Rodríguez y Machuca 2009]. Se han resaltado en gris aquellas áreas en las que interviene el prototipo y el modelo de agentes propuesto.

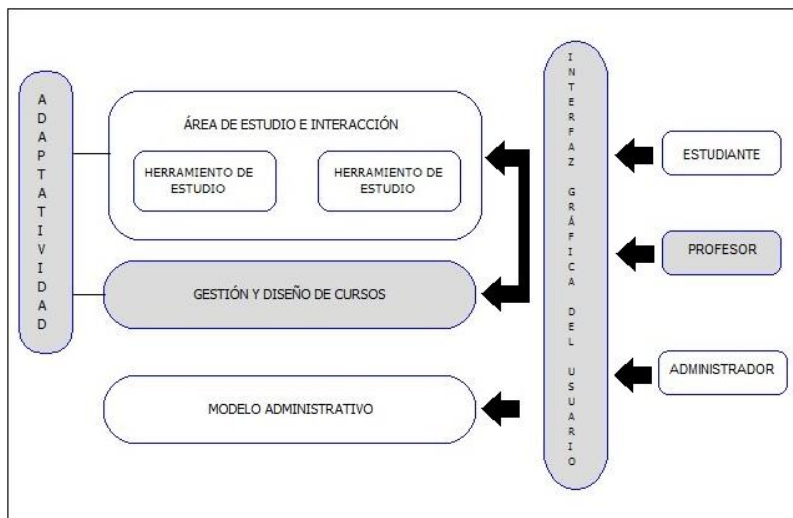


Figura 3. Estructura Conceptual del Sistema [Rodríguez y Machuca 2009]

Los paquetes de Lingweb en los que el modelo de agente interviene se presentan en la Figura 4, estos paquetes son: El paquete PCurso, relacionado con la gestión de cursos, matrícula, gestión de unidades, secuencias, actividades y ejercicios, así como la gestión de material, para el modelo solo se trabajará con la gestión de cursos y la gestión de actividades. El paquete PAdaptatividad encierra todo lo relacionado con los agentes de software diseñados para proveer adaptatividad al modelo de ayudas adaptativas en la plataforma. Finalmente, el paquete GUI(Pag-Web) contiene las interfaces de usuario y lógica de interfaz del sistema [Rodríguez y Machuca 2009].

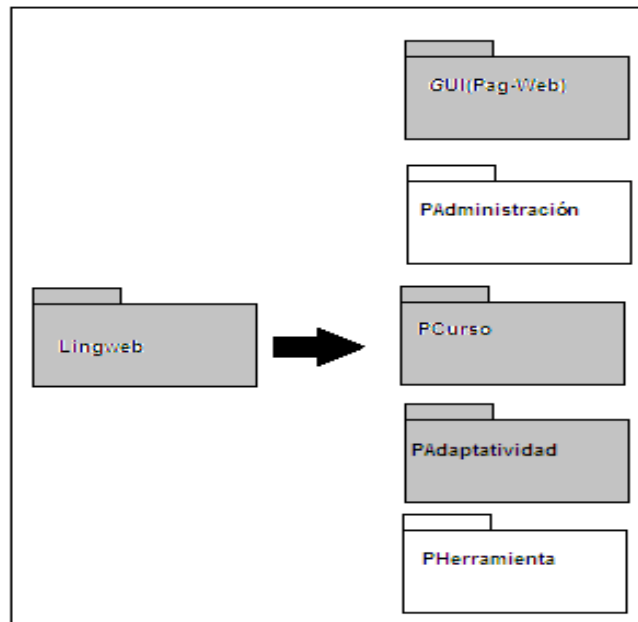


Figura 4. Vista conceptual. [Rodríguez y Machuca 2009]

6.2.5.2 vista lógica

La Figura 5 presenta el modelo distribuido en capas de Lingweb, se visualizan los patrones de diseño controlador y fachada para permitir el control de los eventos de la interfaz del usuario a la lógica, y el acceso de la lógica a los medios de almacenamiento. Se resalta en gris los paquetes donde interviene el modelo de agentes. [Rodríguez y Machuca 2009]

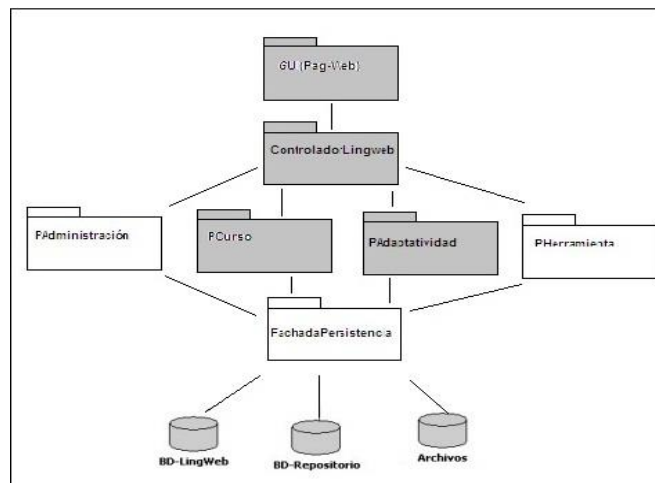


Figura 5. Estructura general por capas [Rodríguez y Machuca 2009]

Del paquete de adaptatividad de Lingweb para se tiene en cuenta los agentes Controlador, Monitor y Ayuda, agentes claves para el desarrollo del modelo de las ayudas adaptativas [Rodríguez y Machuca 2009].

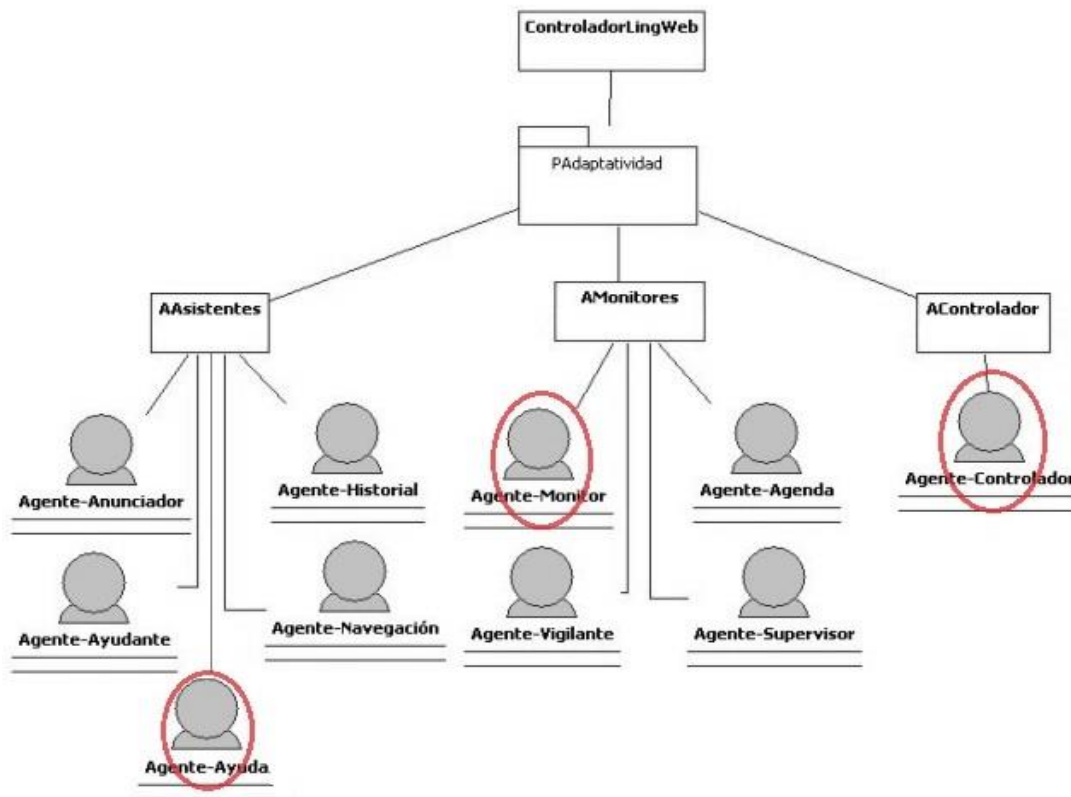


Figura 6. Estructura del Paquete Adaptatividad [Rodríguez y Machuca 2009]

La arquitectura completa de Lingweb se presenta por completo en [Rodríguez y Machuca 2009].

6.3 ESTADO DEL ARTE

A continuación se listan algunos textos académicos, que han sido tomados como base inicial para la estructuración conceptual de la propuesta, intentando buscar proyectos relacionados con *Agentes Software* en el contexto de las *Interfaces Adaptativas* y *sistemas adaptativos*, *los sistemas de ayuda* y *monitoreo de usuarios*.

Methods and techniques of adaptive. Brusilovsky, P. [Brusilovsky 1996]

Este trabajo presenta brevemente una explicación de lo que son los sistemas hipermedia adaptativa, las técnicas, los métodos, las tecnologías de hipermedia adaptativa.

Este documento se refieren a los sistemas de hipermedia adaptativos (SHA) como: “a todos los sistemas de hipertexto e hipermedia que refleja algunas de las características del usuario en el modelo de usuario y la aplicación este modelo para adaptar varios aspectos visibles del sistema al usuario”.

Este artículo se tuvo en cuenta debido a aporta conocimiento teórico acerca de los Sistemas Hipermedia Adaptativos y sobre lo que se puede adaptar, para saber cuál de ellos se adecua más a la herramienta de gestión de ayudas adaptativas propuestas.

ELM-ART: An Intelligent Tutoring System on World Wide Web. Brusilovsky P., Schwarz E., and Weber G. [Brusilovsky, Schwarz & Weber 1996]

Este trabajo describe es un ejemplo de un Sistema de Tutor Inteligente (ITS). ELM-ART proporciona acceso remoto a un material didáctico hipermedia estructurado que incluye explicaciones pruebas, ejemplos y problemas. Lo cual a diferencia de los libros de texto electrónicos, ELM-ART proporciona al alumno apoyo inteligente en la navegación y las posibilidades de jugar con ejemplos para resolver los problemas y para obtener soporte inteligente para resolver problemas, este soporte normalmente suele ser proporcionados por un maestro humano. ELM-ART también puede servir como un buen ejemplo para discutir los problemas de aplicación y portabilidad de los sistemas ITS en la WWW. Los escritores de este artículo aseguran que algunas de las tecnologías ITS, tales como el análisis inteligente de soluciones y la resolución de problema basada en ejemplos se pueden trasladar con relativa facilidad a un contexto de WWW.

Este trabajo se destaca ya que explica un sistema de tutor inteligente, que puede en cierto sentido, servir para que la Herramienta que va a proveer ayudas adaptativas en ambientes e-learning, pueda sugerir ayudas en la navegación, y dar un soporte inteligente a los usuarios.

Adaptive User Interface for Multimedia Communication System Based on Multiagent. Luy, L., Kitagatay, G., Suganumay, T., & Kinoshita, T. [Luy, Kitagatay, Suganumay, & Kinoshita 2003]

Este trabajo ha propuesto la función de visualización de gestión adaptativa que se utiliza para mejorar la flexibilidad de la interfaz de usuario en FVCS (Sistemas de Videoconferencia Flexibles). Mediante la implementación de esta función en FVCS, las preferencias del usuario sobre la GUI se podría obtener con facilidad y el funcionamiento de la ventana se puede realizar de forma adaptativa de acuerdo con las preferencias del usuario y el estado del Sistema Operativo.

Además, el diseño de las funciones de cada agente, permite que la interfaz sea adaptativa, debido a la estructura de la organización del agente, la funcionalidad de cada uno, así como la manera de organización y de colaboración de estos.

Esta aplicación permite que las ventanas de las aplicaciones en FVCS puedan ser configuradas correctamente y de forma automática en respuesta a las preferencias del usuario y el estado del Sistema operativo ya enunciados.

Este trabajo es importante para la función de visualización de gestión adaptativa que se utiliza para mejorar FVCS, y aporta conocimiento de Agentes Inteligentes utilizados para configurar ventanas, que se pueden llegar a utilizar.

User Models in Dialog System, capítulo KNOME: Modeling What the User Knows in UC, Chin, D. N. Paginas 74-107. Springer, Berling, 1989. [Chin 1989]

Knome es el componente de modelado de usuario del Consultor Unix, un sistema de lenguaje natural de ayuda que genera explicaciones para funciones UNIX. Knome utiliza las preguntas del usuario como datos de entrada para calcular la probabilidad de que el usuario pertenece a uno de los cuatro estereotipos globales. Con el modelo de usuario y la dificultad de un concepto, el sistema deduce la probabilidad de que el usuario sabe acerca de un elemento de conocimiento. La adaptación ocurre a través de la salida de la versión de un tema de ayuda correspondiente al estereotipo más probable.

Este trabajo es importante para la adaptación que se utiliza para los mensajes de ayuda, y aporta conocimiento de sobre el manejo de los mensajes de ayuda, que se pueden llegar a utilizar.

Glass Box Approach to Adaptive Hypermedia. PhD tesis, Stockholm University, Höök, K. A, Swedish Institute of Computer Science, 1996. [Höök 1996]

PUSH es una herramienta sensitiva de ayuda-hipermedia para el usuario, que se adapta a la meta de información del usuario. PUSH utiliza un modelo de términos cortos para el objetivo de la información actual del usuario. Cada elemento de ayuda consta de diferentes componentes que apoyan al usuario de diferentes maneras. Dependiendo de la meta de información, el contenido de un elemento de ayuda está adaptado a través de mostrar u ocultar componentes específicos.

Este trabajo es importante para la adaptación que se utiliza para los componentes de la interfaz y aporta conocimiento de sobre el manejo de los mensajes de ayuda enfocándolos a la meta del usuario, que se pueden llegar a utilizar.

Adaptive Help for Webbased Applications. Adaptive Hypermedia and Adaptive Web Based System. Iglesias D. Third International Conference. 2004, pp. 304–307, 2010. [Iglesakis 2010]

El AdaptHelp, trata de combinar los diferentes enfoques presentados anteriormente en Knome y PUSH mediante el uso de métodos de reconocimiento de planes para inferir el conocimiento de los procedimientos del usuario, teniendo en cuenta las especialidades del entorno web.

A diferencia de otros sistemas web adaptativos como ambientes de aprendizaje o los sistemas de recomendación, los sistemas adaptativos de ayuda no puede utilizar los datos de entrada adicionales, como las pruebas o cuestionarios. Por lo tanto sólo los datos de entrada que se pueden recoger, son aquellos que se puedan recoger de forma no intrusiva, es decir, se recogen a partir de los datos que registra las actividades del usuario en el servidor.

Este trabajo es importante ya que aporta conocimiento de sobre el manejo de los mensajes de ayuda enfocándolos a la meta del usuario y los eventos del usuario en el servidor, que se pueden llegar a utilizar.

“A Lightweight Framework for Cross-Application User Monitoring”. Fenstermacher Kurt D, Ginsburng Mark. IEEE. Marzo 2002. [Fenstermacher y Ginsburng 2002]

Framework para la vigilancia de usuarios entre aplicaciones, el framework presentado, monitorea los eventos de más alto nivel para aprender cómo la gente accede, crea y modifica información en lugar de como utilizan las aplicaciones. Este framework combina la tecnología Modelo de Componente de Objetos de Microsoft, Internet Explorer, aplicación Microsoft Office, y del lenguaje de programación Python. Este framework utiliza Modelo de Componente de Objetos (COM) que proporciona la infraestructura que soporta el monitoreo, con COM los programas de Windows pueden controlar las aplicaciones y recibir eventos notificaciones de ellos, con lo que se crea el registro. Al establecer las propiedades e invocar métodos, los desarrolladores pueden utilizar COM para controlar las aplicaciones de a través de un segundo canal, mientras que los usuarios pueden continuar usándolas con normalidad. Se afirma que el framework se puede aplicar para fomentar el entendimiento de dos maneras principales: información sobre el seguimiento de uso de recursos y el análisis de cómo la gente utiliza aplicaciones y herramientas.

Este trabajo es importante para el monitoreo de los usuarios ya que utiliza para los componentes de la interfaz y aporta conocimiento de sobre las manías y hábitos de los usuario en las aplicaciones, que se pueden llegar a utilizar o evitar en éste proyecto.

“A Concept Lattice for Recognition of User Problems in Real User Monitoring”. Oyama Katsunori, Takeushi Atsushi, Ming Hua, Chang Carl K. 18th Asia-Pacific Software Engineering Conference. IEEE 2011 [Oyama, Takeushi, Ming y Chang 2011]

Sistemas de seguridad de intrusos con agentes inteligentes, la seguridad de la red se ha convertido en uno de los aspectos más importantes de los sistemas informáticos y de Internet. El Sistemas de seguridad de intrusos con agentes inteligentes pretende mejorar la seguridad en la red a través de controlar las acciones de los usuarios en tiempo real y para tomar las medidas apropiadas si es necesario. Junto con esto el sistema utiliza predicción a corto plazo para predecir el comportamiento de los usuarios y asesorar al administrador del sistema en consecuencia, antes de que las acciones reales tengan lugar, es decir, pretende informar al administrador sobre acciones sospechosas en la red.

Este trabajo es importante para el monitoreo de los usuarios ya utiliza agentes inteligentes y aporta conocimiento de sobre la predicción a corto plazo del comportamiento de los usuario en las aplicaciones en red, que se pueden llegar a utilizar en éste proyecto.

“An Intelligent Agent Security Intrusion System”. Pikoulas J., W. Buchanan, Mannion M., Triantafyllopoulos K.. Ninth Annual IEEE International Conference Workshop on the Engineering of Computer-Based System (ECBS'02). IEEE 2002. [Pikoulas, Buchanan y Triantafyllopoulos 2002]

El concepto de red de reconocimiento de problemas de los usuarios en el monitoreo de usuario real, los problemas de los usuarios encontrados durante el uso de un producto de software son a menudo difíciles de identificar incluso después de que el producto de software es puesto completamente a prueba y luego liberado. En este estudio de las situaciones orientado al monitoreo de usuarios reales se desarrolla una aplicación de concepto de red para representar los patrones de uso de la interacción del usuario y percibir los problemas potenciales usuarios. La estructura de datos del problema del usuario es modelado basado en la ontología de problema de usuario. Una vez que un problema de usuario es reconocido, los datos de observación de la utilización de patrones son el tema para su posterior análisis.

Este trabajo es importante para el monitoreo de los usuarios y aporta conocimiento de sobre el manejo de problemas de los usuario en las aplicaciones en red, que se pueden llegar a utilizar en éste proyecto.

Para ver por completo el estado del arte de este proyecto remitirse al **ANEXO I - Marco de Referencia**

7. ESPECIFICACIÓN DEL PROTOTIPO DE LA HERRAMIENTA PARA PROVEER AYUDAS ADAPTATIVAS EN AMBIENTES E-LEARNING

Lingweb es un Ambiente virtual basado en una arquitectura multi-agente para la enseñanza y aprendizaje de idiomas, tiene características de un sistema de gestión de cursos en línea diseñado para tres perfiles: administrador, profesor y estudiante. Lingweb como la mayoría de los plataformas e-learning cuenta con el mismo estilo de ayuda para todos los usuarios, por lo cual se tomó como caso de estudio para éste proyecto; dado que presentaba la problemática a la cual se da una posible solución en este proyecto. Con la integración del prototipo Lingweb cuenta una herramienta para proveer ayudas adaptativas para el perfil profesor, al cual se agregó un segundo perfil para clasificar al usuario profesor dependiendo de su conocimiento de Lingweb, en usuario básico, intermedio y avanzado, es decir, usuario profesor-básico, usuario profesor-intermedio y usuario profesor-avanzado, que reciben las ayudas necesarias para el uso de Lingweb. Estas ayudas se brindan sólo

para el formulario de crear curso, los demás formularios no se tuvieron en cuenta por la magnitud de Lingweb y el tiempo del desarrollo del proyecto.

El prototipo de ayudas adaptativas a partir del modelo de agentes software que se integró al sistema multi-agentes de Lingweb, permite la asignación de las ayudas y el monitoreo de los usuarios, tomando en cuenta las características de los usuarios, es decir, según su segundo perfil, su interacción con el aplicación, para ofrecer la ayuda o las ayudas necesarias y los diferentes conocimientos que serán necesarios para su interacción con Lingweb.

7.1 DESCRIPCIÓN Y SELECCIÓN DE LAS AYUDAS QUE OFRECE EL PROTOTIPO

Para el proceso de asignación y creación de las ayudas del modelo de agentes del prototipo se hizo una investigación sobre las ayudas existentes para guiar y asistir a los usuarios en ambientes e-learning.

Las ayudas para brindar guía y asistencia a los usuarios de entornos e-learning [Galitz 2007], se muestran en la Figura 7 y en la Tabla 2 se muestran las ayudas seleccionadas para el modelo de agentes del prototipo.

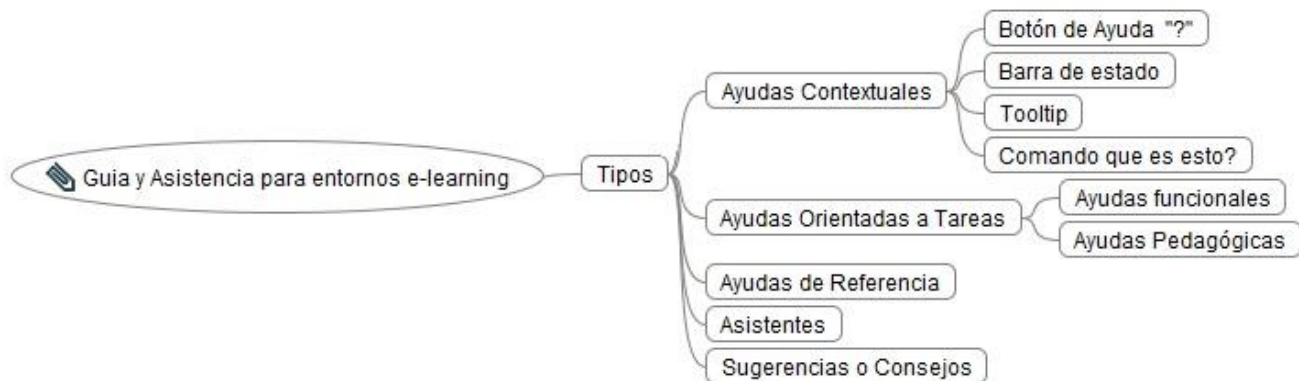


Figura 7. Guía y Asistencia para entornos e-learning del prototipo [Galitz 2007].

Los tipos guía y asistencia en entornos e-learning son:

Estas definiciones se pueden ver en [Galitz 2007].

- Ayuda contextual: proporciona información en el contexto de una tarea que se realiza, o alrededor de un objeto específico que se está operando. Los tipos comunes de Ayuda contextual incluyen botones de comando la ayuda, los mensajes de la barra de estado e información sobre herramientas, el tooltip
- La ayuda Orientado a tareas: a veces llamado la ayuda de procedimiento, se presenta en una ventana principal, su propósito es detallar procedimientos para llevar a cabo una tarea. Las ayudas pedagógicas, no se tuvieron en cuenta para este proyecto, pues estas se salen del contexto del trabajo.
- Sugerencias o Consejos: es un botón de comando situado en una pantalla que proporcionan algunos elementos importantes del contexto, pero concreta la información relacionada con la que se muestra en pantalla.
- La ayuda de Referencia: es otra forma de documentación en línea; su propósito es presentar información de ayuda que puede ser referencia orientada, documentando las características de un producto, o puede servir como guía de un usuario a un producto.
- Asistentes: Un asistente es un conjunto estructurado de pantallas que guían al usuario a través de un proceso de entrada de la toma de decisiones o de datos. Los asistentes se muestran en una ventana secundaria. Las pantallas incluyen controles para recoger la entrada, y los botones de navegación de

mando situados en la parte inferior de página (Atrás, Siguiente, Finalizar y Cancelar). Un asistente se suele acceder a través de botones de la barra de herramientas o iconos. Como se mencionó en el prototipo no se crearon asistentes. Los asistentes no se tuvieron en cuenta para este proyecto, pues estas se salen del contexto del trabajo.






Lingweb como la mayoría de las plataformas e-learning contaba con el mismo estilo de ayuda para todos los usuarios, las cuales en su inicio eran manual de usuario, botones de comando la ayuda, los mensajes de la barra de estado, el tooltip, es decir, sólo contaba con ayudas contextuales; según la descripción anterior de las ayudas. Para el prototipo las ayudas seleccionadas fueron:

- Ayuda contextual: a partir del manual de usuario, botones de comando la ayuda (a partir de los iconos del agente ayuda, y el icono de sonido), los mensajes de la barra de estado, el tooltip.
- La ayuda Orientado a tareas: a partir del Comando ¿Qué es esto?, que Microsoft Windows ha introducido.
- Sugerencias o Consejos: a partir del comando ¿Sabias Qué?
- La ayuda de referencia: a partir de ventanas con la ayuda sobre el formulario de creación de cursos.

Dado que el prototipo sólo es para el usuario profesor, al cual se le agregó un segundo perfil para clasificar al usuario profesor dependiendo de su conocimiento de Lingweb, en usuario básico, intermedio y avanzado, es decir, usuario profesor-básico, usuario profesor-intermedio y usuario profesor-avanzado, que recibe las ayudas necesarias para el uso de Lingweb según éste perfil, como se mencionó anteriormente, para garantizar que el usuario profesor de Lingweb obtenga las ayudas necesarias según su segundo perfil al que llamaremos **rol_ayuda**, se definieron las ayudas que recibe cada perfil de **rol_ayuda**, de la siguiente manera:

- Usuario Básico: este usuario profesor-básico recibe todas las ayudas creadas: las ayudas contextuales, las ayudas orientadas a tareas, sugerencias o consejos y la ayuda de referencia, según sea necesario.
- Usuario Intermedio: este usuario profesor-Intermedio recibe las ayudas creadas: las ayudas contextuales, las ayudas orientadas a tareas y la ayuda de referencia.
- Usuario Avanzado: este usuario profesor-Avanzado recibe las ayudas creadas: las ayudas contextuales y la ayuda de referencia.

A continuación se muestra en la Tabla 2, el **rol_ayuda** y las ayudas que crea el agente ayuda (Los agentes se mencionaran más adelante en el modelo de agentes).

En la tabla 2 se puede ver la relación entre el perfil de usuario y las ayudas del prototipo, marcando con una x las ayudas que recibirá cada usuario dependiendo de su perfil. La diferencia entre las ayudas contextuales del usuario básico, intermedio y avanzado, marcado en la Tabla 2 con una “x” en el usuario avanzado, consiste en que el usuarios básicos e intermedio tendrán todas las ayudas contextuales, es decir, cuenta con el manual de usuario, los mensajes de la barra de estado, el tooltip y en el caso del botón de ayuda, en Lingweb al utilizar el icono o botón de ayuda , el agente ayuda crea la ayuda necesaria para el usuario a partir de la creación de dos iconos, uno el icono del agente  y el otro el icono de sonido , mientras que el usuario avanzado cuenta con con el manual de usuario, los mensajes de la barra de estado, el tooltip y en el caso de utilizar el icono o botón de ayuda , el agente ayuda crea la ayuda necesaria para el usuario a partir de la creación del icono del agente .

ROL_AYUDA (Segundo perfil del usuario profesor)	TIPOS DE AYUDA			
	Contextuales	Referencia	Orientadas a Tareas	Sugerencias o Consejos
Usuario Básico	x	x	x	x
Usuario Intermedio	x	x	x	
Usuario Avanzado	x	x		

Tabla 2. Tipos de ayudas del modelo

Como se asignan y se crean las ayudas por parte del agente ayuda se explicará más adelante.

7.2 DESCRIPCIÓN ENFOQUE PARA LA ASIGNACIÓN Y CREACIÓN DE LAS AYUDAS DEL PROTOTIPO

El prototipo de la herramienta para proveer ayudas adaptativas en ambientes e-learning se creó pensándolo como un sistema de ayuda adaptativo que se encarga de la asignación y creación de las ayudas adaptativas, para lo cual se hizo una investigación sobre los sistemas de ayuda, los sistemas inteligentes y los sistemas de ayuda adaptativos para definir el enfoque que se utilizó en el prototipo.

Los sistemas de ayuda para aplicaciones informáticas son programas que asisten al usuario en su utilización. Por esta razón su objetivo principal es el de proporcionar respuestas a los problemas que se presentan en el uso de la aplicación [Iglesakis 2004].

Existen diversas maneras de ofrecer asistencia o ayuda a los usuarios, unas de ellas son los sistemas de ayuda inteligente [Breuker 1990], [Winkels 1992] y los sistemas de ayuda adaptativos [Iglesakis 2004]. Los sistemas de ayuda inteligente (SAI) o asistentes inteligentes, son una propuesta para mejorar el aprendizaje y el rendimiento en la utilización de entornos informáticos complejos. Habitualmente la ayuda tiene como único fin la mejora del rendimiento del usuario, por ejemplo, en el Sistema de Ayuda Inteligente Aran o SAI Aran, se plantea que la asistencia debe tener también un propósito educativo. Si el usuario mejora su conocimiento sobre la aplicación su necesidad de ayuda decrecerá con el tiempo. Un punto clave de los SAI es el énfasis que se hace en la adaptación de la ayuda proporcionada a las necesidades y los conocimientos específicos de cada usuario [Fernández 2001].

Los sistemas de ayuda adaptativos usan principalmente dos enfoques para adaptar la información de ayuda a los distintos requisitos de información de los diferentes usuarios. El primer enfoque es inferir el conocimiento del usuario mediante la observación de la interacción del usuario con el sistema de ayuda y el segundo enfoque es considerar la situación real de un usuario a través de un plan de reconocimiento. Dentro de estos sistemas

de ayuda adaptativa se encuentran KHOME, que infieren el conocimiento de los usuarios, y PUSH, que utilizan reconocimiento de planes para inferir el plan real o información objetivo del usuario [Iglesakis 2004].

Tomando en cuenta estas definiciones sobre los sistemas de ayuda se definió para el prototipo el enfoque de inferir el conocimiento del usuario mediante la observación de la interacción del usuario con el sistema, para así ofrecer las distintas ayudas adaptativas.

7.3 DESCRIPCIÓN Y SELECCIÓN DEL MÉTODO DE MONITOREO DE USUARIO DEL PROTOTIPO

Existen diversas razones para monitorear a los usuarios en [Fenstermacher y Ginsburng 2002], se presenta un framework para la vigilancia de usuarios entre aplicaciones para mejorar la interacción hombre-máquina; en [Oyama, Takeushi, Ming y Chang 2011], se muestra un sistemas de seguridad de intrusos con agentes inteligentes, cuyo objetivo es mejorar la seguridad en red a través del monitoreo y control de los usuarios, por últimos en [Pikoulas, Buchanan y Triantafyllopoulos 2002], se presenta el concepto de red de reconocimiento de problemas de los usuarios en el monitoreo de usuario real, este pretende identificar las fallas en el uso de un producto de software a partir del monitoreo de usuarios. La razón para el monitoreo de los usuarios en el prototipo como se mencionó en la sección 7.2, la cual es poder inferir el conocimiento del usuario mediante la observación de la interacción del usuario con el sistema, para lo cual se necesita saber la actividad que está realizando o que desea realizar el usuario para brindarle la ayuda necesaria para realizarla.

Existen diversos métodos para el monitoreo de usuarios, cuyo objetivo principal es hacer un seguimiento de los usuarios en diversas aplicaciones que pueden ser web, o de escritorio. Algunos sistemas lo hacen con objetivos benéficos como lo hacen Moodle [Kuldeep y Poonphon 2008], [Fenstermacher y Ginsburng 2002] se presenta un framework para la vigilancia de usuarios, [Oyama, Takeushi, Ming y Chang 2011] se muestra un sistemas de seguridad de intrusos con agentes inteligentes, [Pikoulas, Buchanan y Triantafyllopoulos 2002] se presenta el concepto de red de reconocimiento de problemas de los usuarios en el monitoreo de usuario real y otros no tantos como los keylogger [Sagiroglu y Canbek 2009].

Existen diversos usos para el monitoreo de usuarios como [Kuldeep y Poonphon 2008], cuyo objetivo es medir el avance de los usuarios en plataformas virtuales de aprendizaje y [Fenstermacher y Ginsburng 2002] se presenta un framework para la vigilancia de usuarios, [Oyama, Takeushi, Ming y Chang 2011] se muestra un sistemas de seguridad de intrusos con agentes inteligentes, [Pikoulas, Buchanan y Triantafyllopoulos 2002] se presenta el concepto de red de reconocimiento de problemas de los usuarios en el monitoreo de usuario real que se explican mejor en la sección 6 en el marco de referencia, en el estado del arte. A continuación se hace una breve descripción de los sistemas y sus métodos de monitoreo investigados para la definición del método de monitoreo de usuarios del prototipo:

Los keyloggers son conocidos indistintamente como software de seguimiento, programas informáticos actividad de vigilancia, los registradores de pulsaciones, rastreadores de teclado, y snoopware. Aunque el principal objetivo de los keyloggers es para supervisar las acciones de un usuario del teclado, que ahora tienen capacidades que van más allá de esa función. Estos pueden realizar un seguimiento de prácticamente cualquier cosa que se ejecuta en un ordenador. Los keyloggers avanzados pueden realizar un seguimiento de cosas tales como las operaciones de cortar, copiar y pegar, el uso de Internet, las operaciones de archivo (ejecutar, crear, renombrar, modificar y borrar), y las impresiones. Los keyloggers se usan también para el control de las "conductas de los usuarios", y para recopilar información, como información de identificación personal o privada. Los keyloggers son diferentes de otros tipos de spyware o malware, como virus y gusanos. Ellos comparten los recursos del sistema (por ejemplo, CPU y memoria), con programas legítimos, permanecen

residentes en el sistema de forma invisible por el tiempo que sea necesario, y son cuidadosos y diseñados simplemente para hacer sus tareas, sin llamar la atención de los usuarios. Este tipo de monitoreo se hace invisible para el usuarios final que los hace que sean considerados nocivos, pues al ser usados por personas indiscriminadas pueden terminar afectado al usuario pues permiten la obtención de cualquier información sea contraseñas de correos, archivos privados, etc. [Sagioglu y Canbek 2009]

Moodle es una plataforma de licencia de software libre. La palabra Moodle es un acrónimo de Modular Object-Oriented Dynamic Learning Environment. Todos los involucrados en el negocio del e-learning también lo llaman como un Sistema de Gestión de Aprendizaje (LMS) o Entorno Virtual de Aprendizaje (EVA). Moodle está diseñado para ayudar a los educadores a crear cursos en línea con las oportunidades de una rica interacción. La interactividad se está convirtiendo en un aspecto clave del e-learning. Moodle registra todas las actividades, incluyendo visitas y mensajes para todos los objetos de aprendizaje organizadas en el sistema y ofrece "Informes" y "estadísticas" para ayudar a los expertos en el tema a mejorar la calidad de los cursos e-learning. Este monitoreo pretende ayudar a mejorar los sistema e-learning permitiendo ver el avance de los estudiante y este monitoreo solo es en la plataforma e-learning y no de todo el sistema como lo son los keylogger. [Pikoulas, Buchanan y Triantafyllopoulos 2002]

El prototipo tiene un enfoque para inferir el conocimiento del usuario mediante la observación de la interacción del usuario con el sistema, para así ofrecer las distintas ayudas adaptativas. La observación se hará a través del monitoreo de usuarios cuyo método consiste en hacer un seguimiento de los eventos que representan una acción del usuario en la plataforma, se hace a través del monitoreo de los eventos del mouse sobre los iconos y botones, y los eventos en los formularios a través de las acciones dadas en los campos de texto que contiene la interfaz gráfica.

En la siguiente sección se explicarán los métodos para monitoreo de los usuarios y la asignación de las ayudas del prototipo.

7.4 DESCRIPCIÓN DE LOS MÉTODOS DE MONITOREO DE USUARIO Y DE ASIGNACIÓN DE AYUDAS DEL PROTOTIPO

Para el prototipo de herramienta de ayudas adaptativas se creó un métodos para el monitoreo de usuarios y otro método para la asignación de las ayudas, que se describirán comenzando con el método de monitoreo seguido del método de asignación de ayudas. La figura 8, muestra la apariencia de la plataforma Lingweb.



Figura 8. Apariencia de Lingweb.

Para el prototipo se crearon los agentes monitor y ayuda que se integraron al agente pasarela que ya existía en la arquitectura multiagente de Lingweb descrita en la sección 6.2.5.

Breve descripción de los servicios de los Agentes del prototipo

Para el prototipo se crearon tres agentes que son:

- **Agente pasarela:** es el encargado de comunicar los eventos de parte del usuario al agente monitor y de mostrar la ayuda asignada por el agente ayuda.
- **Agente monitor:** es el encargado de recibir los mensajes del agente pasarela sobre los eventos por parte del usuario, el agente monitor los almacena ya sea en la base de datos o en el archivo de log de usuarios, una vez tiene el mensaje decide si debe pedir ayuda para el usuario al agente ayuda, también es el encargado de determinar si un usuario está activo o inactivo a partir de los mensajes que recibe del agente pasarela.
- **Agente ayuda:** es el encargado recibir las peticiones de ayuda del agente monitor, el agente ayuda hace la asignación de la ayuda y la crea y se la envía al agente pasarela para que este se la muestre al usuario en la interfaz y el agente ayuda registra la ayuda enviada en la base de datos o en el log de usuarios.

7.4.1 MÉTODO DE MONITOREO DE USUARIOS

El método de monitoreo de usuarios que se plantea consiste en monitorear todos los eventos que representa una acción del usuario en la plataforma, se hace a través del monitoreo de los eventos del mouse sobre los iconos y botones, y los eventos en los formularios a través de los eventos sobre los campos de texto que contiene la interfaz gráfica en el paquete GUI. Este proceso se hace por parte de los agentes monitor y pasarela.

Cabe aclarar que solo se tuvo en cuenta el formulario de crear cursos de la plataforma Lingweb, en cuanto al monitoreo de los campos de textos y botones en ellos. En cuanto al monitoreo de eventos del mouse sobre los iconos, enlaces y botones de la interfaz que pertenecen a las herramientas de trabajo de Lingweb situados en el lado izquierdo de la imagen contará con todos estos iconos, enlaces.

El método de monitoreo de usuarios se hace a través de todos los eventos que representan una acción del usuario en la plataforma, es decir, se capturarán los eventos del mouse que haga el usuario en la interfaz, ya sea en los iconos, en los enlaces, en los botones y en los campos de texto de los formularios, a continuación se describe el algoritmo para el monitoreo de usuarios en los eventos del mouse sobre los iconos, enlaces y botones y de los campos de texto de formulario del formulario de crear cursos.

7.4.1.1 Algoritmo para el Monitoreo de los eventos del mouse sobre los iconos, enlaces y botones

Para explicar mejor el monitoreo de los eventos sobre los iconos, enlaces y botones en las figuras 9 y 10, se muestran las imágenes de los iconos, enlaces y botones de Lingweb que se monitorean. En la figura 9 se pueden apreciar los iconos y enlaces pertenecientes a Herramientas Generales o General Tool (icono de libreta de apuntes, foro, chat, multimedia), Herramienta de Curso o Course Tool (iconos de cuaderno, enlaces, diccionario, cuaderno, diario, publicaciones, cartelera y notificaciones), lista de cursos o Course list (con los enlaces de los cursos existentes, curso 12, 22222, curso JLA 004 y curso solo actividades), repositorio (con los enlaces de general repository y course repository) y por ultimo notificaciones (con el icono notificaciones) de izquierda a derecha de la figura 9.

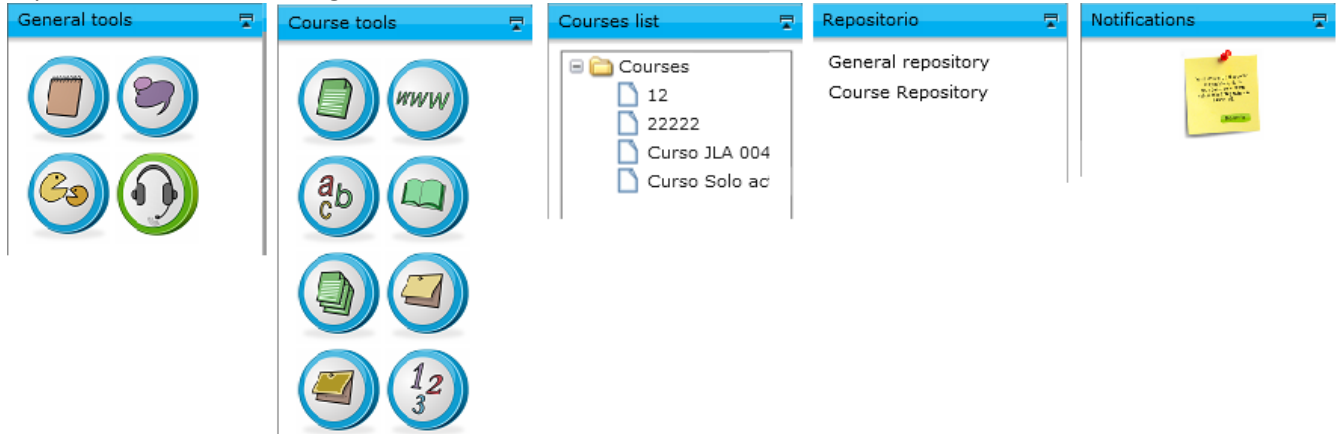


Figura 9. Icono, enlaces de las herramientas de Lingweb que monitorea el prototipo

En la figura 10 se muestran los botones pertenecientes al formulario de crear cursos que se monitorearan, que son los iconos de ayuda, el botón save o guardar, el botón del historial, en botón de cancel o cancelar y el botón de preview o vista previa, de izquierda a derecha en la figura 10.



Figura 10. Botones del formulario de cursos de Lingweb que monitorea el prototipo

El algoritmo de monitoreo de usuarios necesita de los agentes pasarela, monitor y ayuda que se describen en el modelo de agentes en la siguiente sección 7.5, para el prototipo se crearon los agentes monitor y ayuda que se integraron al agente pasarela que ya existía en la arquitectura multiagente de Lingweb descrita en la sección 6 del Marco de referencia. Para el prototipo se tuvieron en cuenta los cinco casos de los eventos de los usuarios.

7.4.1.1.1 Cinco casos de los eventos de los usuarios

El algoritmo de monitoreo se creó para funcionar en cinco posibles casos de los eventos de los usuarios, que son clic o doble clic en los iconos o enlaces, clic derecho en los enlaces, clic botones del formulario en especial el icono de ayuda de los campos de texto, clic en los campo de texto del formulario de crear curso y por último en caso de que no se reciban de eventos del usuario.

7.4.1.1.1.1 clic o doble clic en los iconos o enlaces

Crearemos la tabla de eventos de clic o doble clic en los iconos de course tool, de los enlaces de course list y repositorio a partir la figura 9. Como se muestra en la tabla 3.

En la tabla en la columna de iconos o enlaces, se muestra el icono o el enlace asociado al evento del mouse. El nombre que tiene el icono o el enlace. El evento del mouse se captura el evento, el id del usuario que hizo el evento y la hora en la que lo hizo. Mensaje que se envía es la palabra icono o enlace seguido del código del evento, la hora del evento y el id del usuario que hizo el evento, este mensaje se envía como mensaje del evento al agente monitor.

Una vez el agente monitor recibe mensaje verifica que el evento sea de tipo icono o enlace si es de este tipo lo almacena en la base de datos, esto es lo que hace el agente monitor con este tipo de eventos.






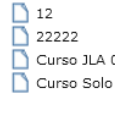
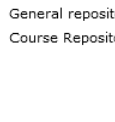
ICONOS	NOMBRE ICONO O ENLACE	EVENTO DEL MOUSE	MENSAJE QUE SE ENVIARÁ
	cuaderno	Clic o doble clic sobre el icono	Icono, HT_Cu, Hora, id_user
	enlaces	Clic o doble clic sobre el icono	Icon, HT_Ww, Hora, id_user
	glosario	Clic o doble clic sobre el icono	Icono, HT_Gl, Hora, id_user
	diccionario	Clic o doble clic sobre el icono	Icono, HT_Dic, Hora, id_user
	diario	Clic o doble clic sobre el icono	Icono, HT_Dia, Hora, id-user
	LC_enlace	Clic o doble clic sobre los enlaces	Enlace, LC_enlace, Hora, id_user
	R_enlace	Clic o doble clic sobre los enlaces	repositorio, R_enlace, Hora, id_user

Tabla 3. Eventos sobre los iconos y enlaces

7.4.1.1.1.2 clic derecho en los enlaces

El agente pasarela es el encargado de comunicar los eventos de los usuarios, si el evento es de tipo clic derecho en los enlaces, se lo envía al agente monitor como un mensaje de eventos de la siguiente manera enlaceD, nombre del enlace, el id del usuario que hizo el evento, y la hora. Para explicar mejor el comportamiento de agente monitor se tendrá en cuenta la lista de cursos de la figura 11.

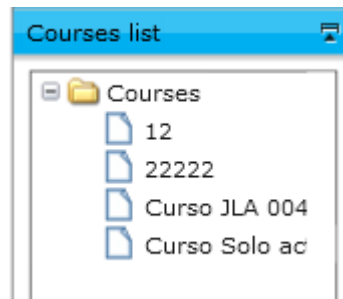




Figura 11. Lista de cursos de Lingweb

El agente monitor recibe el mensaje y si es del repositorio lo almacena en la base de datos, si el evento es de lista de cursos, el busca en la base de datos si hay algún clic en  Courses si no se encuentra ningún evento de clic sobre este icono, se busca en la base de datos si hay más eventos de clic derecho sobre los enlaces de lista de curso de encontrarse, el agente monitor registra el evento en la base de datos y pide ayuda al agente ayuda para este tipo de eventos (se explicará con más detalle en el algoritmo de asignación de ayudas). Si por el contrario en la base de datos hay algún clic en  Courses entonces el agente monitor registra el evento en la base de datos.

7.4.1.1.1.3 clic en los campo de texto del formulario de crear curso

Para explicar el monitoreo de los campos de texto del formularios de crear cursos la figura 12, muestra uno de los formularios de Lingweb para el caso de clic en los campos de texto del formulario de crear curso. El formulario de crear cursos cuenta con los campos de texto para id, code, group, name, language, credits, academic period, activation date, work domain, type, guest, authoring history, y el check box cloneable course, los cuales tienen sus propios iconos de ayuda, como se muestra en la figura 12.

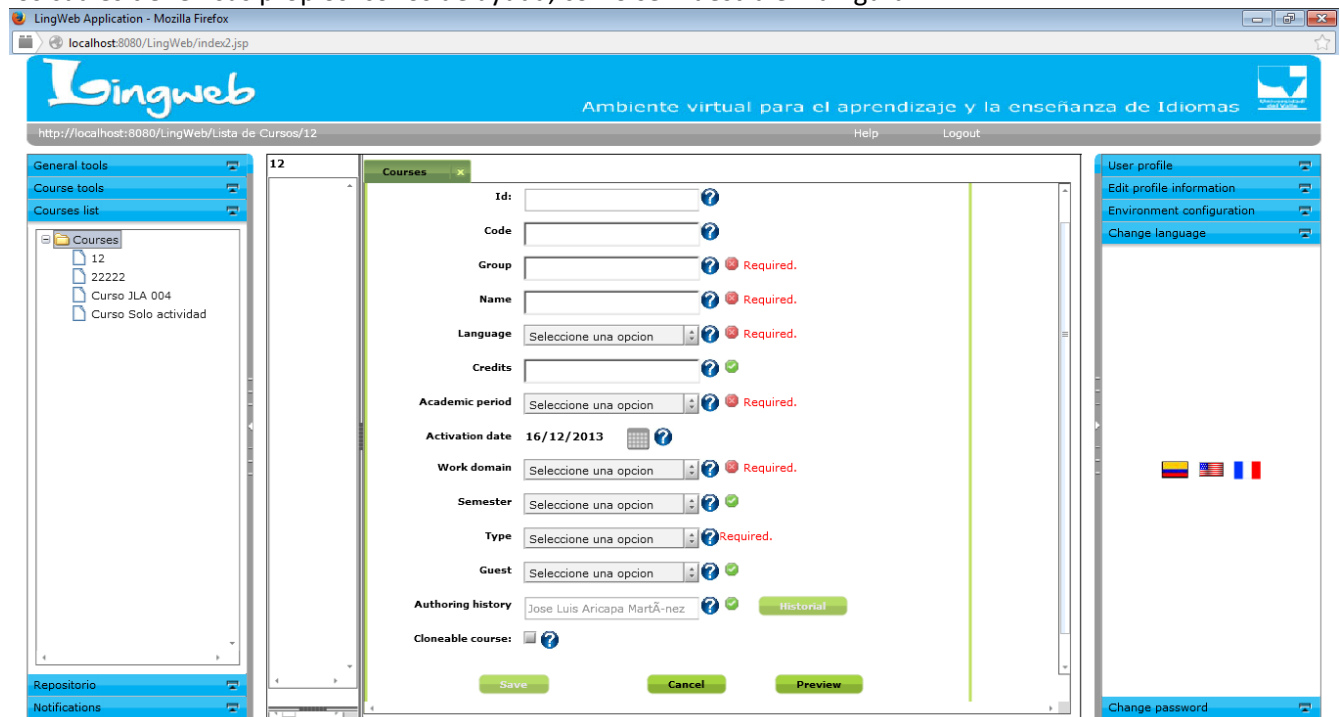


Figura 12. Formulario crear Curso Lingweb

Crearemos la tabla de eventos a partir la figura 12, como se muestra en la tabla 4.

En la tabla 4, El IDENTIFICADOR UNICO será el identificador que tiene el campo de texto. EVENTO DEL MOUSE será la captura del identificador única al momento en el que el usuario de clic sobre este campo de texto, el agente pasarela será el encargado del envío del mensaje. MENSAJE QUE SE ENVIARÁ será un código, junto con la hora del evento, el id del usuario que hizo el evento, que se envía como mensaje al agente monitor.

IDENTIFICADOR UNICO	EVENTO DEL MOUSE	MENSAJE QUE SE ENVIARÁ
Id	Clic sobre el campo de texto.	Texto, LC_FCC_I, HORA, id_user
Código	Clic sobre el campo de texto.	Texto, LC_FCC_cod, HORA, id_user
Grupos	Clic sobre el campo de texto.	Texto, LC_FCC_Gr, HORA, id_user
Nombre	Clic sobre el campo de texto.	Texto, LC_FCC_Nom, HORA, id_user

Tabla 4. Eventos sobre los campos de texto

El agente pasarela envía este mensaje al agente monitor, el agente monitor si el evento es de los campos de texto él lo almacena en la base de datos.

7.4.1.1.1.4 clic en los botones del formulario en especial el icono de ayuda de los campos de texto

El agente pasarela es el encargado de comunicar los eventos de los usuarios, si el evento es de tipo clic en los botones o en el icono de ayuda del formulario de crear curso mostrados en la figura 10, el agente pasarela envía un mensaje de evento de la siguiente manera botón, nombre del botón, el id del usuario que hizo el evento, y la hora. Si el evento es del tipo clic o doble clic sobres el botón save o guardar, el botón del historial, en botón de cancel o cancelar y el botón de preview o vista previa el agente monitor lo almacenará en la base de datos, si por lo contrario el evento es en el icono de ayuda de la figura 10, el agente monitor envía el mensaje pidiendo ayuda para ese campo de texto junto al icono de ayuda.

7.4.1.1.1.5 en caso de que no se reciban eventos del usuario

El agente monitor tiene un tiempo determinado para cada usuario en el que no hay ningún mensaje de eventos por parte del agente pasarela, una vez se cumpla ese tiempo que es para el usuario básico 1 minuto, para el usuarios intermedio un minuto y medio y para el usuario avanzado dos minutos, el agente monitor revisa la base de datos para ver que usuarios ya sobrepasaron ese tiempo y una vez tenga el id del usuario y la ultimo evento que realizo, le pide ayuda al agente ayuda para este usuario. Si después de enviada la ayuda (se explicará en el método de asignación de ayudas) no hay ningún evento por parte del usuario se asume que el usuario está inactivo y no envía más ayudas hasta que haya un nuevo evento.

Para finalizar podemos notar que el algoritmo de monitoreo en general consiste en la captura de los eventos del mouse sobre los componentes de la interfaz gráfica de usuario de la plataforma Lingweb, lo que se busca es clasificar los mensajes que se envían por el evento sobre cada componente, para de esta forma no causar que se brinden las ayudas equivocadas a cada componente como se verá en el método de asignación de ayudas.

7.4.2 MÉTODO DE ASIGNACIÓN DE AYUDAS

Antes de aclarar el método de asignación de ayudas debemos especificar las ayudas a utilizar para brindar asesoría a los usuarios están descritas y seleccionadas en la sección 7.1 descripción y selección de las ayudas que ofrece el prototipo. En la tabla 2, se muestran los tipos de ayudas con las que cuenta el prototipo y los perfiles de usuario con los que se cuenta. Lingweb cuenta con los perfiles de usuario Administrador, Docente y Estudiante.

Las ayudas que ofrece el prototipo son de dos tipos ayudas invocadas por el usuario, es decir, aquellas que ofrece el prototipo a partir de los iconos de ayuda o del link de ayuda; y las ayudas ofrecidas por el sistema como respuesta al monitoreo de las acciones de los usuarios como son la ayuda por referencia, la ayuda del comando *qué es esto?*, la ayuda *sabías qué?*.

Dado que el prototipo sólo es para el usuario profesor, al cual se le agregó un segundo perfil para clasificar al usuario profesor dependiendo de su conocimiento de Lingweb, en usuario básico, intermedio y avanzado, es decir, usuario profesor-básico, usuario profesor-intermedio y usuario profesor-avanzado, que recibe las ayudas necesarias para el uso de Lingweb según éste perfil, como se mencionó anteriormente, para garantizar que el usuario profesor de Lingweb obtenga las ayudas necesarias según su segundo perfil al que llamaremos **rol_ayuda**, se definieron las ayudas que recibe cada perfil de **rol_ayuda**, de la siguiente manera:

- Usuario Básico: este usuario profesor-básico recibe todas las ayudas creadas: las ayudas contextuales, las ayudas orientadas a tareas, sugerencias o consejos y la ayuda de referencia, según sea necesario.
- Usuario Intermedio: este usuario profesor-Intermedio recibe las ayudas creadas: las ayudas contextuales, las ayudas orientadas a tareas y la ayuda de referencia.
- Usuario Avanzado: este usuario profesor-Avanzado recibe las ayudas creadas: las ayudas contextuales y la ayuda de referencia. El prototipo trabaja con el segundo perfil para asignar las ayudas y solo cuenta con el perfil docente o profesor.

Teniendo en cuenta cuales ayudas provee el prototipo se explicará a continuación el algoritmo para asignar las ayudas.

7.4.2.1 Algoritmo para para la asignación de las ayudas

El algoritmo para la asignación de las ayudas necesita de los agentes pasarela, monitor y ayuda que se describen en el modelo de agentes en la sección 7.5. Para el prototipo se tuvieron en cuenta los cinco casos de los eventos de los usuarios.

En el primer paso el agente monitor le envía un mensaje al agente para los siguientes casos de eventos del usuario:

- clic derecho en los enlaces
- clic en los botones del formulario en especial el icono de ayuda de los campos de texto
- en caso de que no se reciban eventos del usuario

El agente ayuda recibe el mensaje de parte del usuario monitor, lo procesa y dependiendo del evento asigna la ayuda y la crea para enviársela al agente pasarela quien es el encargado de mostrársela al usuario. A continuación se describen y muestran las ayudas creadas para los tres casos de eventos mostrados anteriormente (clic derecho en los enlaces, clic en los botones en especial en el icono de ayuda, en el caso de que no se reciban eventos del usuario):

7.4.2.1.1 Ayuda para el caso clic derecho en los enlaces

Si el evento es de lista de cursos, y se elige la opción de crear cursos el agente ayuda crea la ayuda sabias qué?, se la envía al agente pasarela y este se la muestra al usuario. En la figuras 13 se muestra las opciones para la creación de cursos, creación de cursos a partir de clonación y eliminar cursos, de la herramienta de crear curso y en la figura 14 se muestra la ayuda sabias qué?, que muestra el agente pasarela, si es un usuario básico o intermedio, si es el usuario avanzado no se asigna ninguna ayuda.

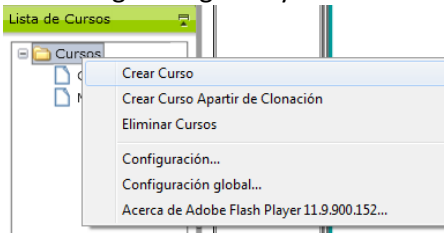


Figura 13. Opción de crear cursos de Lingweb

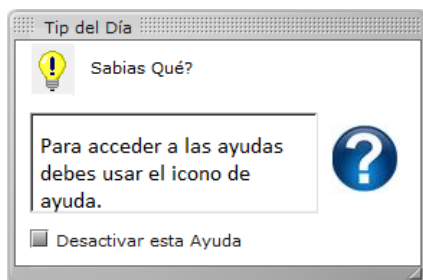


Figura 14. Ayuda sabias qué?, creada por el agente ayuda

Si el evento es de la lista de curso, se hace en varias ocasiones y no se elige la opción de crear cursos, el agente ayuda asigna y crea la ayuda del comando qué es esto? y se la envía al agente pasarela para que se la muestre al usuario, si es un usuario básico o intermedio, si es el usuario avanzado no se asigna ninguna ayuda. En la figura 15 se muestra la ayuda de comando qué es esto?, creada por el agente ayuda.

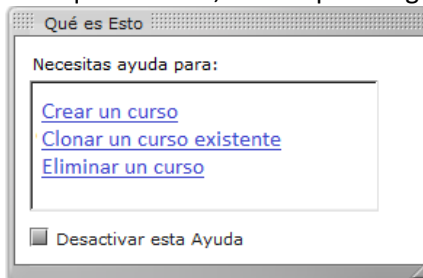







Figura 15. Ayuda comando qué es esto?, creada por el agente ayuda

7.4.2.1.2 Ayuda para el caso clic en los botones del formulario en especial el icono de ayuda de los campos de texto

Si hay un evento en el icono de ayuda , en el formulario de creación de cursos, el agente monitor le envía el mensaje pidiendo ayuda al agente ayuda, quien asigna y crea la ayuda contextual del ayudante de Lingweb y se lo envía al agente pasarela para que se lo muestre al usuario. Al utilizar el icono o botón de ayuda , el agente

ayuda crea la ayuda necesaria para el usuario a partir de la creación de dos iconos, uno el icono del agente  y el otro el icono de sonido  si es un usuario básico o intermedio, si es un usuario avanzado sólo se crea el icono del agente , al dar clic en el icono del agente, el agente ayuda crea la ayuda contextual del ayudante de Lingweb. En la figura 16 se muestra la ayuda creada si se da clic en el icono de ayuda del campo de texto idioma y se da clic en el icono del agente, el agente ayuda creará la siguiente ayuda.

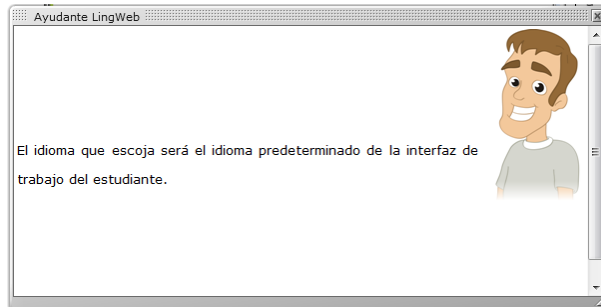


Figura 16. Ayuda contextual ayudante creada por el agente ayuda

7.4.2.1.3 Ayuda para el caso de que no se reciban eventos del usuario

Si no hay evento después de pasado los tiempo que es para el usuario básico 1 minuto, para el usuarios intermedio un minuto y medio y para el usuario avanzado dos minutos en el algoritmo de monitoreo y el agente monitor envía el mensaje pidiendo ayuda el agente ayuda asigna y crea la ayuda de referencia que se muestra en la figura 17.

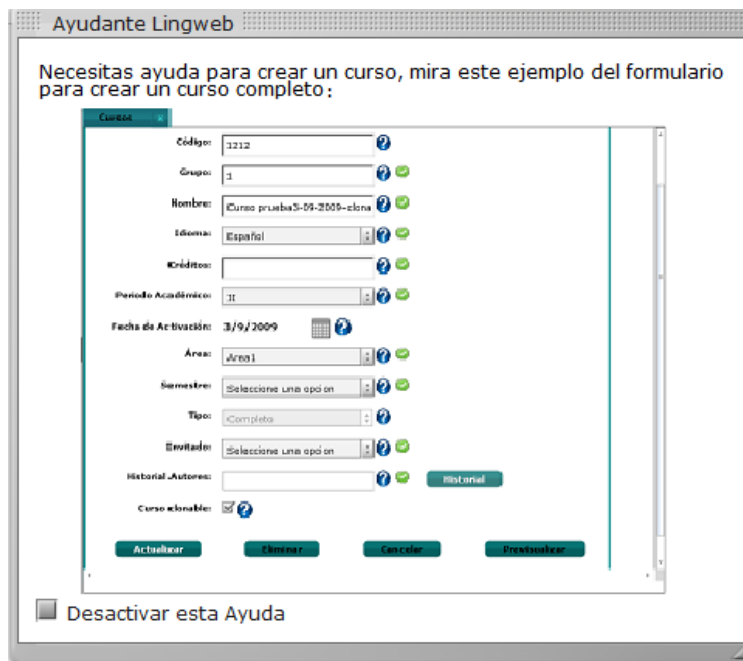


Figura 17. Ayuda de referencia creada por el agente ayuda

El método de asignación de ayudas asegura que las ayudas sean las correctas y que brinden al usuario la cantidad de información necesaria para llevar a cabo sus tareas en la plataforma.

7.5 MODELO DE AGENTES PARA PROVEER AYUDAS ADAPTATIVAS DEL PROTOTIPO

Teniendo en cuenta los modelos de la metodología INGENIAS [INGENIAS 2005], se diseñó el sistema multiagente, estos modelos son:

7.5.1 MODELO DE OBJETIVOS

Este modelo refleja los objetivos que debe cumplir el prototipo. En la Figura 18 se presentan el objetivo general del prototipo y en las Figuras 19 y 20 se definen los subobjetivos sugeridos al descomponer el objetivo general.

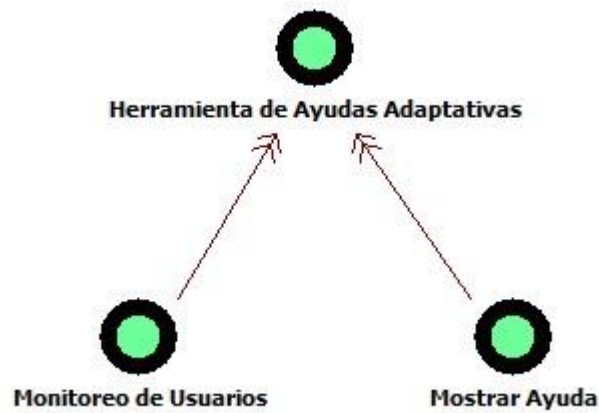


Figura 18. Modelo de Objetivo General



Figura 19. Modelo de Sub-objetivo Monitorear Usuarios

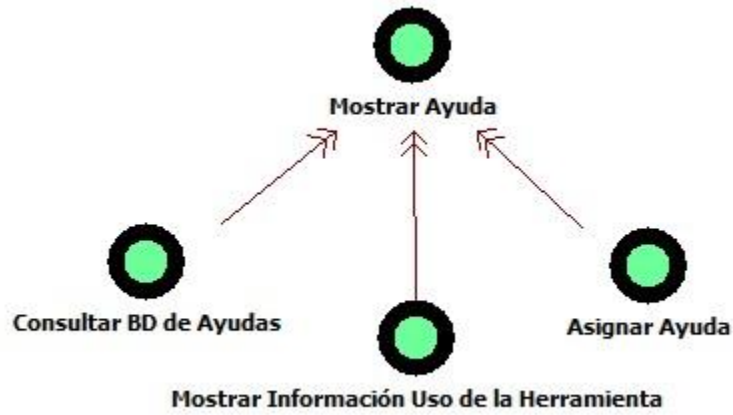


Figura 20. Modelo de Sub-objetivo Mostrar Ayuda

7.5.2 MODELO DE AGENTES

Este modelo describe cómo son los agentes y los objetivos asociados a ellos. En particular describe su funcionalidad, la asignación de los roles que juega y las tareas que deberán ejecutar. En la Figura 21 se muestra el modelo del agente ayuda y el agente monitor.

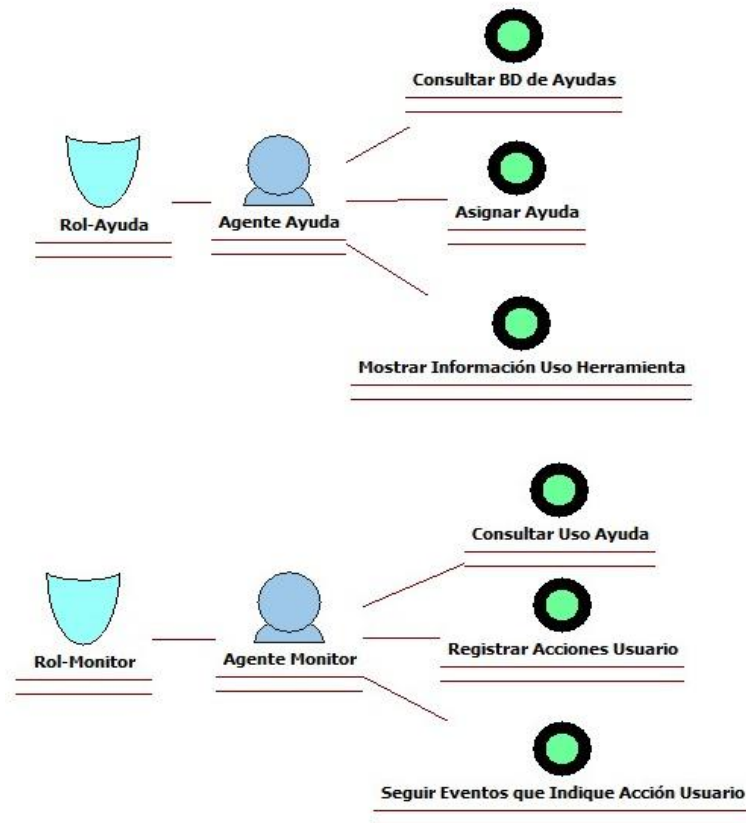


Figura 21. Detalle del Modelo de Agentes Monitor y Ayuda

En la Tabla 5 se describen las responsabilidades de los agentes con el fin de dar mayor claridad de sus funcionalidades.

AGENTE	RESPONSABILIDAD	DESCRIPCIÓN
PASARELA (AGENTE CONTROLADOR)	<ol style="list-style-type: none"> 1. Comunicar servlets con agentes. 2. Crear agentes 3. Eliminar agentes 	<p>Este agente es el encargado de comunicar los servlets con el sistema multiagente, cualquier tipo de información o petición que se desee enviar u obtener de algún agente en especial o de varios agentes debe pasar a través del agente pasarela.</p> <p>Crear el agente monitor y un agente ayuda una vez ingresa un usuario al sistema y lo destruye o elimina una vez sale del mismo.</p>
MONITOR	<ol style="list-style-type: none"> 1. Seguir eventos que indiquen una acción de un usuario. (seguir eventos del mouse, monitorear los campos de los formularios). 2. Consultar tablas de ayudas que ha utilizado. 	<p>Este agente tiene la finalidad de monitorear los movimientos de los usuarios para saber qué está haciendo en determinado momento y de acuerdo con lo que están haciendo comunicárselo al agente ayuda.</p>
AYUDA	<ol style="list-style-type: none"> 1. Consultar tablas en la Base de Datos. 2. Asignar ayuda. 3. Mostrar ayuda (mostrar información del uso de la herramienta). 	<p>Este agente tiene la finalidad de ofrecer la ayuda, el agente monitor le envía mensajes sobre lo que está haciendo el usuario en determinado momento y de acuerdo a lo que están haciendo muestra la ayuda.</p>

Tabla 5. Responsabilidades de los agentes

7.5.3 MODELO DE ORGANIZACIÓN

El modelo de organización es comparable a la estructura del sistema de agentes. En él se ofrece una visión de las relaciones existentes entre las entidades activas (agentes y roles), las pasivas (aplicaciones y recursos) y los elementos que proporcionan la funcionalidad del sistema (flujos de trabajo y tareas). La Figura 22 muestra la el modelo de organización de Lingweb al que se le agregó el prototipo resaltando en óvalos rojos los agentes del prototipo, el objetivo general de Lingweb es **dotar de adaptatividad al sistema**.

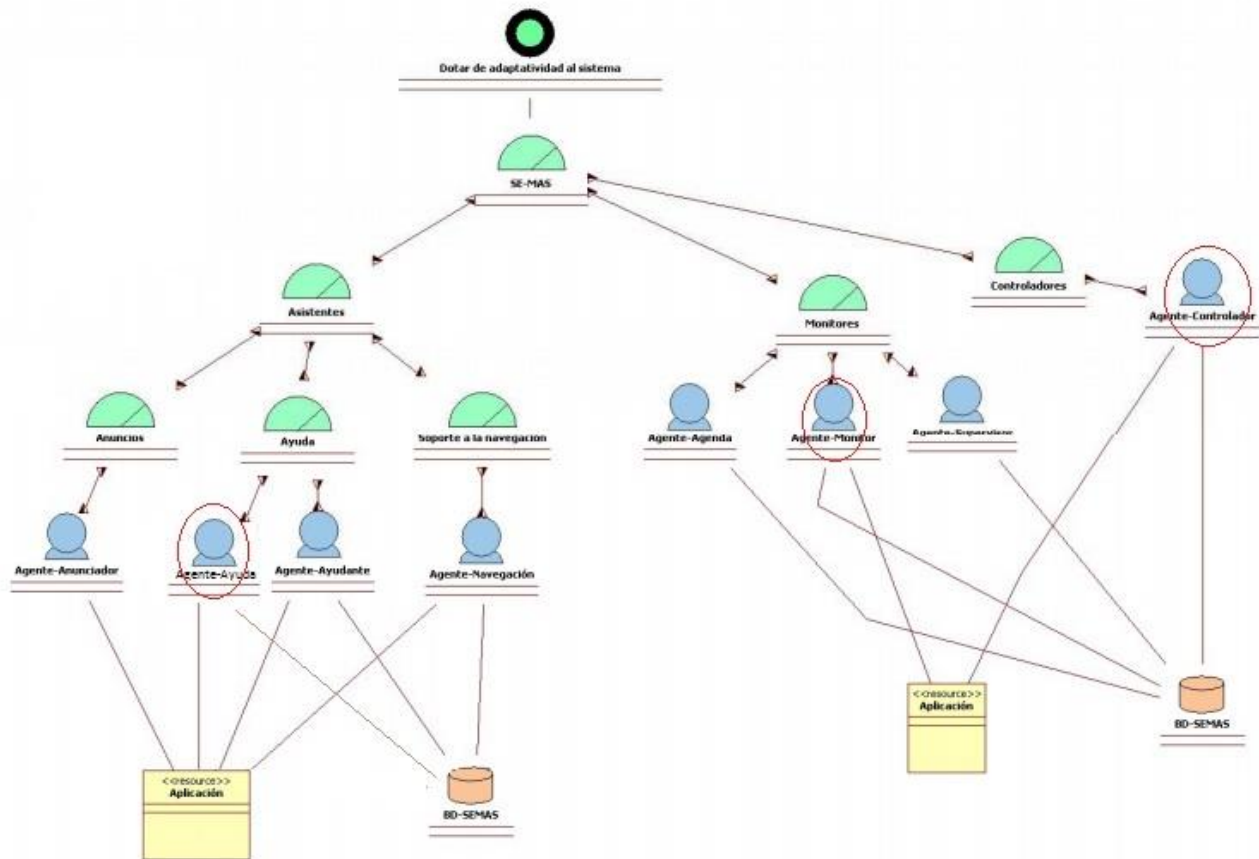


Figura 22. Modelo de Organización de Lignweb

7.5.4 MODELO DE ENTORNO

Este modelo especifica las entidades del entorno en el que se encuentra el sistema de agentes, ya sean recursos consumibles o no consumibles, aplicaciones, o agentes. En la Figura 23 se muestra el modelo de entorno del prototipo, en él aparecen todos los agentes del sistema y los recursos que se utilizan a lo largo de su ejecución.

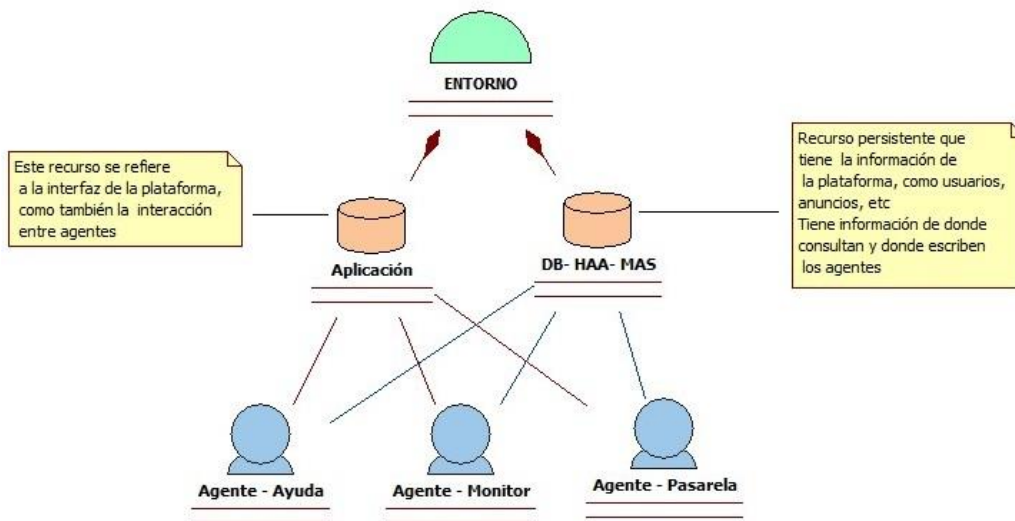


Figura 23. Modelo de Entorno del prototipo

7.5.5 MODELO DE SERVICIOS

En este modelo se especifican las funciones y servicios prestados por el agente individualmente. El modelo de conocimiento es un refinamiento del modelo de interacción, que define además los enlaces de comunicación entre los agentes.

- Modelo de Servicios: comunicar eventos del usuario

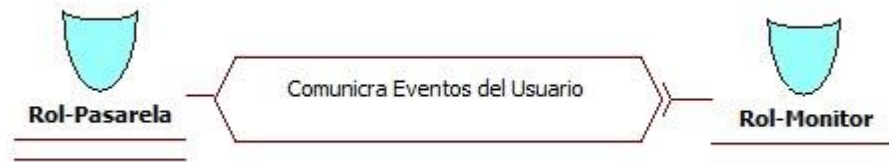


Figura 24. Modelo de Servicios: comunicar eventos del usuario

- Modelo de Servicios: pedir ayuda

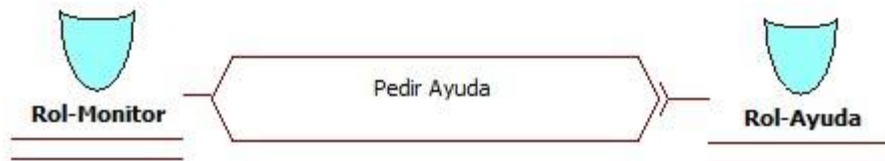


Figura 25. Modelo de Servicios: pedir ayuda

- Modelo de Servicios: mostrar ayuda

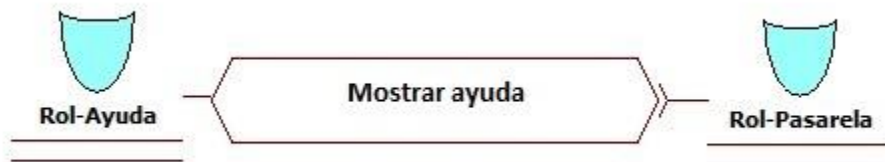


Figura 26. Modelo de Servicios: mostrar ayuda

Para ver Modelo de Agentes del Prototipo completo remitirse al **ANEXO III Modelo de agentes y casos de uso**.

8. DESARROLLO PROTOTIPO

Dado que el propósito de este proyecto es la creación de la herramienta para proveer ayudas adaptativas en ambientes e-learning, se desarrolló el prototipo en la plataforma Lingweb. A continuación se describen: la metodología de desarrollo, tecnologías utilizadas, el análisis de requerimientos, la implementación, diseño e implementación del prototipo.

8.1 METODOLOGÍA DE DESARROLLO

La metodología utilizada fue una metodología RUP Ágil, donde los artefactos generados se hicieron en base al documento SRS (Software Requirements Specification) en lugar de utilizar historias de usuarios, ya que el documento SRS permite un mayor control en la especificación de requerimientos. En general, las etapas que conforman dicha metodología son: Análisis, Diseño, Implementación y pruebas del sistema. El desarrollo de las aplicaciones interactivas fue dividido en diferentes fases, y en ellas se definieron, de acuerdo con los requerimientos especificados en el SRS (Ver Anexo I – Software Requirements Specifications), actividades a realizar en cada una como se muestra en la Tabla 6.

FLUJO DE TRABAJO	TIEMPO EN MESES								
	1	2	3	4	5	6	7	8	9
Fase de Inicio (Análisis) <ul style="list-style-type: none"> Acercamiento a la plataforma. Análisis de requerimientos. 									
Fase de diseño <ul style="list-style-type: none"> Modelo de navegación. Modelo de persistencia. Interfaz de usuario. Diseño del modelo de ayudas. 									
Fase implementación <ul style="list-style-type: none"> Código fuente. Ejecutable. Manual de usuario. 									
Fase pruebas <ul style="list-style-type: none"> Documento que describe las pruebas unitarias, de integración 									
<ul style="list-style-type: none"> Documentación del proyecto 									

Tabla 6. Fases del proceso de desarrollo en meses

Dependiendo de las actividades de trabajo de cada fase, se estableció las iteraciones que se muestran en la tabla 7. Cada iteración está compuesta por el análisis, diseño, implementación, ejecución de pruebas, integración y planeación de la próxima iteración. Para cada grupo de actividades se realizaron las iteraciones, cuya duración de un mes se dividió en tiempos de 4 o 5 días más o menos.

FLUJO DE TRABAJO	TIEMPO EN MEDIAS SEMANAS							
	1	2	3	4	5	6	7	8
Análisis								
Diseño								

Implementación								
Ejecución de pruebas								
Iteración								
Planeación de la próxima iteración								

Tabla 7. Iteración un mes se dividió en tiempos de 4 o 5 días más o menos

8.2 TECNOLOGÍAS UTILIZADAS

En el desarrollo del proyecto se continuó con las herramientas tecnológicas que se manejan desde la primera etapa de Lingweb, estas definiciones se tomaron de [Aricapa y Puentes 2009], las tecnologías son:

OpenLaszlo

OpenLaszlo es una plataforma de código libre XML-nativo para la creación de aplicaciones Web. Las aplicaciones resultantes son totalmente compatibles con los principales navegadores del mercado y sistemas operativos. Permite crear como archivo final un objeto Flash, formato SWF. Este framework está pensado para desarrollar aplicaciones multimedia o componentes multimedia con los mismos resultados que el software de pago de Adobe/Macromedia Flash. Con la particularidad de que la escritura de código no es la utilizada en Adobe/Macromedia Flash (ActionScript), sino que es un lenguaje de programación basado en XML-nativo orientado a objetos y Javascript, con todas las ventajas que esto supone. Las posibilidades son casi infinitas como podemos ver en la propia web de OpenLaszlo, en las que podemos ver cómo han implementado gestores de correo electrónico, plataformas de E-commerce y demás aplicaciones. [Openlaszlo 1], [Openlaszlo 2]

Java

Lenguaje de programación orientado a objetos desarrollado por Sun Microsystems. Sun describe el lenguaje Java como “simple, orientado a objetos, distribuido, orientado, robusto, seguro, de arquitectura neutra, portable, de altas prestaciones, multitarea y dinámico”. Con Java se pueden escribir programas convencionales como para Internet. Se caracteriza por ser independiente de la plataforma, lo que lo pone en ventaja frente a otros lenguajes de programación. Esto significa, que el código producido por el compilador Java puede transportarse a cualquier plataforma (Intel, Sparc, Motorota, etc.) que tenga instalada una máquina virtual Java y ejecutarse.

JPA

Java Persistence API (JPA) proporciona un estándar para gestionar datos relacionales en aplicaciones Java SE o Java EE, de forma que además se simplifique el desarrollo de la persistencia de datos. [Eclipse 2]

J2EE

Java Platform, Enterprise Edition o Java EE (anteriormente conocido como Java 2 Platform, Enterprise Edition o J2EE hasta la versión 1.4), es una plataforma de programación parte de la Plataforma Java para desarrollar y ejecutar software de aplicaciones en lenguaje de programación Java con arquitectura

de niveles distribuida, basándose ampliamente en componentes de software modulares ejecutándose sobre un servidor de aplicaciones. La plataforma Java EE está definida por una especificación. Similar a otras especificaciones del Java CommunityProcess, Java EE es también considerada informalmente como un estándar debido a que los suministradores deben cumplir ciertos requisitos de conformidad para declarar que sus productos son conformes a Java EE.

JSP (Java Server Pages)

Tecnología orientada a crear páginas Web con programación en Java. El motor de las páginas JSP está basado en los servlets de Java (programas en Java destinados a ejecutarse en el servidor). Con JSP se puede crear páginas de manera parecida a como se crean en ASP o PHP (otras dos tecnologías de servidor). Se Generan archivos con extensión .jsp que incluyen, dentro de la estructura de etiquetas HTML, las sentencias Java a ejecutar en el servidor. Antes de que sean funcionales los archivos, el motor JSP lleva a cabo una fase de traducción de esa página en un servlet, implementado en un archivo class (Byte codes de Java). Esta fase de traducción se lleva a cabo habitualmente cuando se recibe la primera solicitud de la página .jsp, aunque existe la opción de precompilar en código para evitar ese tiempo de espera la primera vez que un cliente solicita la página.

Log4j

Log4j es una biblioteca open source desarrollada en Java por la Apache Software Foundation que permite a los desarrolladores de software elegir la salida y el nivel de granularidad de los mensajes o “logs” (logging) a tiempo de ejecución y no a tiempo de compilación como es comúnmente realizado. La configuración de salida y granularidad de los mensajes es realizada a tiempo de ejecución mediante el uso de archivos de configuración externos. [Log4j]. Ver ANEXO VII - Manual de Instalación

JADE

Jade es un middleware que proporciona tanto un entorno de desarrollo como un entorno de ejecución para la realización y mantenimiento de sistemas multiagente. El entorno de desarrollo está formado por una serie de librerías en Java que permiten la implementación de agentes de manera limpia e independiente de la plataforma sobre la que se va a ejecutar. Cumple con las especificaciones de FIPA. Permite la comunicación entre agentes a través del paso de mensajes ACL. Este framework está implementado completamente en el lenguaje JAVA. [JADE]

PostgreSQL

PostgreSQL es un sistema de gestión de bases de datos objeto-relacional (ORDBMS) de software libre manejado por una comunidad de desarrolladores y organizaciones comerciales las cuales trabajan en su desarrollo. [Postgres]

Eclipse

Eclipse es un entorno de desarrollo integrado de código abierto multiplataforma. Dispone de un Editor de texto con resaltado de sintaxis. La compilación es en tiempo real. Tiene pruebas unitarias con JUnit, control de versiones con CVS, integración con Ant, asistentes (wizards) para creación de proyectos, clases, tests, etc., y refactorización.

Asimismo, a través de "plugins" libremente disponibles es posible añadir control de versiones con Subversion e integración con Hibernate y JPA. El producto se encuentra bajo licencia Eclipse Public License (EPL) [Eclipse].

Servicios REST

Este tipo de servicios utiliza el protocolo http para el envío y recepción de peticiones a la capa lógica. Las peticiones al servidor podrán ser POST, GET, PUT y DELETE (las operaciones típicas del HTTP). La respuesta del servidor siempre será una cadena XML con el resultado de la consulta. Ventajas de REST

- Ligero: no hace falta mucho XML de configuración.
- Resultados legibles.
- Fácil de implementar: no hacen falta herramientas específicas.

8.4 ANÁLISIS DE REQUERIMIENTOS

En esta etapa se utilizó la norma IEEE-830 (SRS) dado que permite expresar de manera formal el funcionamiento que se espera del sistema.

Para propósitos del SRS, los usuarios de la plataforma Lingweb, se definieron como usuarios: profesor, estudiante y administrador, pero el del prototipo de ayudas solo tendrá al usuario docente o profesor e implícitamente al sistema, como se muestra en la Figura 18:



Figura 27. Usuarios Lingweb y usuario del prototipo de ayudas

Dependiendo de su perfil, el usuario puede realizar las siguientes funciones descritas a continuación en la Tabla 8:

ACTORES	FUNCIONES
Profesor	<ul style="list-style-type: none"> • Acceder a las ayudas • Desactivar las ayudas

Sistema	<ul style="list-style-type: none"> • Permitir a los usuarios tener diversas ayudas para desarrollar tareas determinadas en la aplicación. • Monitorear las actividades de los usuarios y las ayudas en la aplicación. • Gestionar las ayudas (encargado de determinar la ayuda a brindar), según el perfil del usuario. • Mostrar las ayudas.
---------	---

Tabla 8. Funciones de las aplicaciones distribuidas por perfil de usuario

Se ha establecido el formato de la Tabla 9 para especificar cada uno de los requerimientos:

ESTRUCTURA DE REQUERIMIENTOS	
Título	Muestra el número y el nombre del requerimiento
Función	Se describe lo que debe realizar el requerimiento
Descripción	Se describe detalladamente lo que debe cumplir el requerimiento
Entradas	Se muestra la información necesaria para cumplir el requerimiento
Fuentes	Se describe el modo que se utiliza para ingresar la información
Salidas	Muestra la respuesta que debe generar el sistema
Proceso	Pasos que debe seguir el usuario para ejecutar la funcionalidad
Restricciones	Reglas a cumplirse para la ejecución de la funcionalidad de un requerimiento.
Precondiciones	Reglas a tener en cuenta antes de ejecutar un proceso.
Poscondiciones	Reglas a aplicar después de ejecutar un proceso.
Efectos Colaterales	Cambios que se deben aplicar después de ejecutar un proceso.
Tipo	Prioridad del requerimiento.

Tabla 9. Estructura de los requerimientos en el SRS.

R 1.1 Monitorear de usuarios	
Función	Permitir el monitoreo de los usuarios en la plataforma
Descripción	Consiste en monitorear todos los eventos que representa una acción del usuario en la plataforma, se hará a través del monitoreo de los eventos del mouse sobre los iconos y botones, y los eventos en los formularios a través de los eventos sobre los campos de texto que contiene la interfaz gráfica.
Entradas	Captura del evento.
Fuentes	Mouse.
Salida	Se almacena el evento en el log del usuario.
Proceso	El usuario da clic sobre cualquier icono, botón o campo de texto en la interfaz gráfica, el sistema almacenará la acción del usuario en el log del usuario.
Restricciones	
Precondiciones	El usuario se debe haber autenticado en el sistema.
Poscondiciones	
Efectos colaterales	El sistema internamente guardará el evento en el log de usuarios.
Fecha	24/06/2012
[Tipo]	Primario y esencial

REALIZADO	FIRMA	FECHA
Luz Adriana Peñaranda Guevara		24/06/2012
APROBADO	FIRMA	FECHA
Javier M. Reyes V.		09/07/2012

Tabla 10. Requerimiento creación ejercicio de apareamiento.

Para ver en totalidad los requerimientos de este proyecto, remitirse al **Anexo I– SRS Software Requirements Specifications**.

8.5 MODELO GENERAL DE INTEGRACION ENTRE SERVLETS Y JADE DEL SISTEMA MULTIAGENTE DE LINGWEB

El sistema multiagente de la aplicación Lingweb se apoya en JADE (Java AgentDevelopmentFrameWork). Jade es una de las plataformas de agentes más utilizadas y robustas, que permite una adecuada integración con Lingweb. Para establecer comunicación entre la plataforma y Lingweb se necesita del paquete que proporciona Jade, `jade.wrapper.gateway` que contiene las clases `JadeGateway`, `GatewayAgent`, `atewayBehaviour`, que son las clases que realizan el enlace con los Servlets de Lingweb y el sistema multiagente [Aricapa y Puentes 2009], [Machuca 2009]. En la figura 8 se puede observar una idea general de la comunicación entre Servlets y los Agentes (Jade), que utiliza el sistema multiagentes de Lingweb al que se agregó el prototipo.

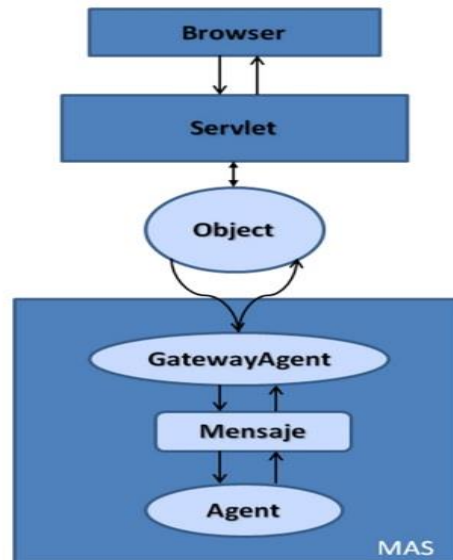


Figura 28. Modelo general de integración Servlets y Jade [Aricapa y Puentes 2009].

8.6 DISEÑO

Lingweb se encuentra desarrollado bajo una arquitectura de tres capas: Modelo Vista y Controlador.

La vista, en donde recae la parte de la presentación o interfaz gráfica es manejada por OpenLaszlo. El modelo o la parte lógica y persistencia de datos, es desarrollada en java. La comunicación entre estas dos capas es realizada por medio de servicios Web denominados Servicios REST. Los estándares de creación de los componentes que comunican cada una de dichas capas, se detalla en la Figura 29 [Aricapa y Puentes 2009].

Para la continuidad de la arquitectura del proyecto, el prototipo se desarrolló bajo el mismo modelo Vista Controlador.

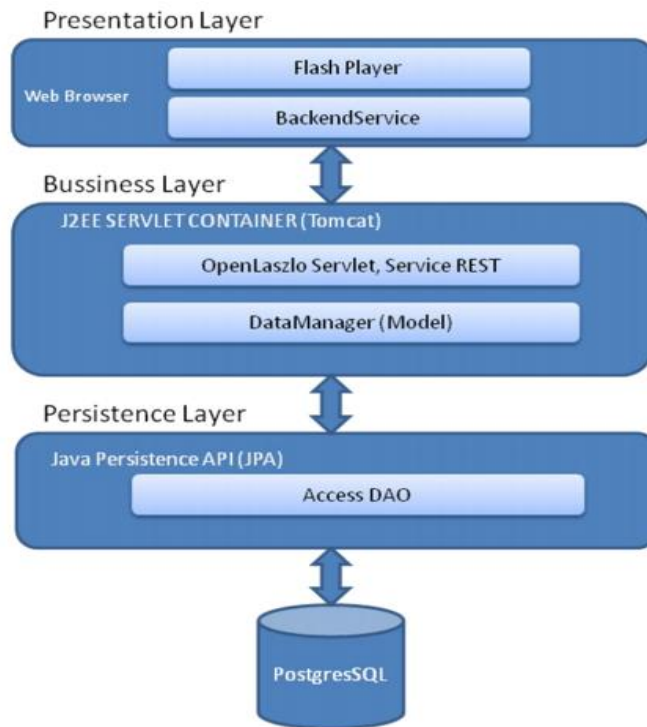


Figura 29. Arquitectura LingWeb en tres capas [Aricapa y Puentes 2009]

8.6.1 DISEÑO COMPUTACIONAL

Esta etapa comprende la elaboración de modelos de navegación, diagrama de clases y diagrama de Persistencia para la parte Web del Sistemas. Adicionalmente, se elaboraron los modelos de organización, entorno, agentes, interacción y comportamientos para los agentes del prototipo que dio soporte a la parte adaptativa, que se describe en la sección 7.5 Modelo de Agentes.

8.6.1.1 Modelo de navegación

El modelo de navegación define y estructura el orden de ejecución de las diferentes funcionalidades que realiza el sistema cuando el usuario interactúa con la aplicación. La figura 30 muestra el modelo de navegación del monitoreo de usuario para el caso de **clic en los campo de texto del formulario de crear curso**, al hacer clic en los campos del formulario dependiendo del campo se le envía un mensaje a monitor a través de la clase MonitorRest y como es la opción de campos de texto del formulario la clase MonitorRespose almacena el mensaje en la base de datos.

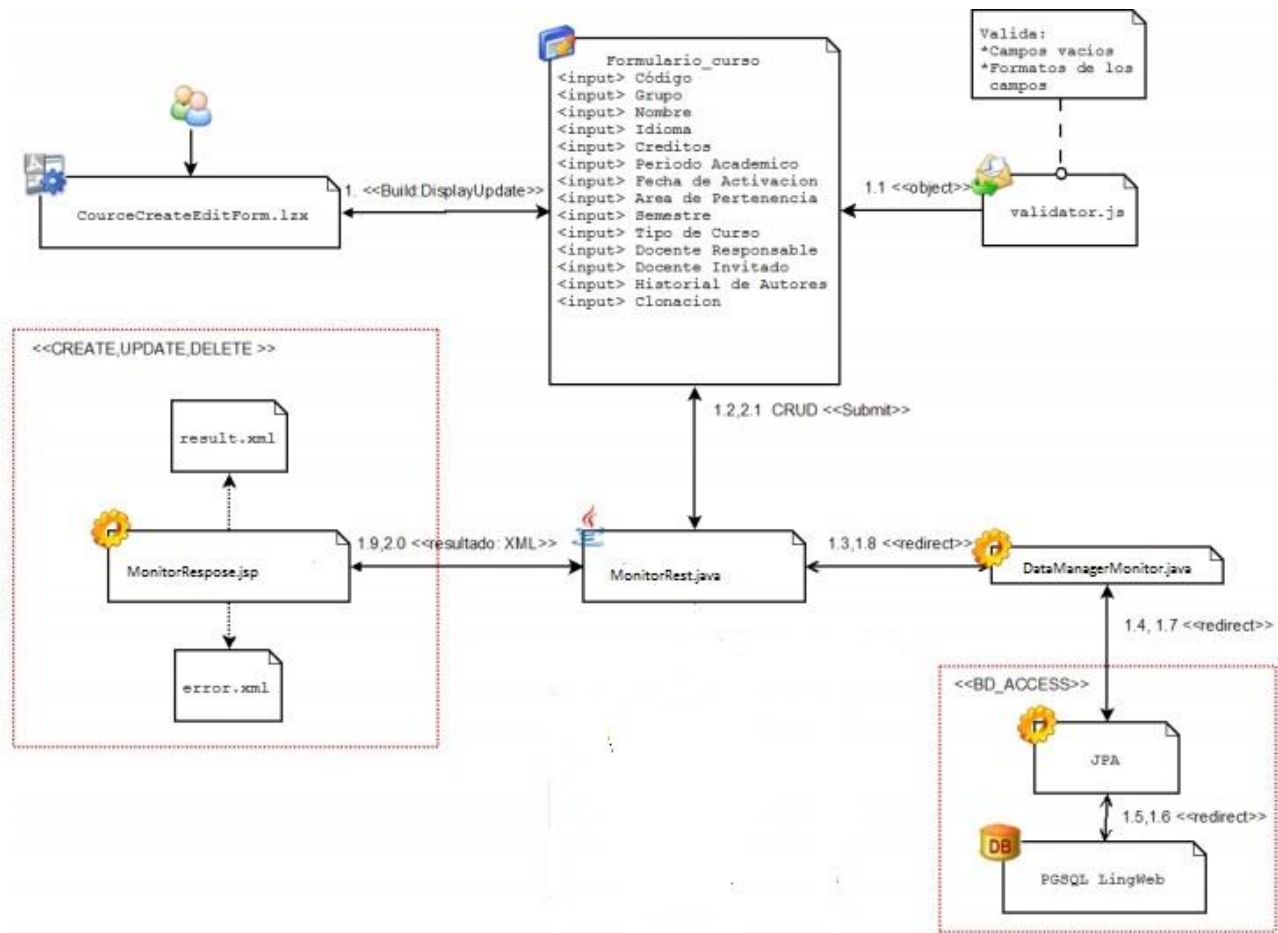


Figura 30. Modelo de navegación del monitoreo de usuarios

8.6.1.2 Diagrama de clases

Los diagramas de clases permiten conocer de manera técnica, la estructura de un sistema y en este caso la estructura de la estructura del monitoreo de usuarios, mostrando las clases que lo componen, sus relaciones con otras clases, atributos y métodos.

Las clases fueron creadas bajo el esquema y los estándares de las clases ya existentes en Lingweb. Además, se tuvo en cuenta el estándar perteneciente a cada capa de la arquitectura del proyecto: presentación, lógica y persistencia.

Para la creación del prototipo se utilizó la parte de persistencia para la creación de las ayudas, ya que los mensajes están almacenados en la base de datos, también se utilizó para el almacenamiento de los eventos del usuario.

A continuación, en la figura 31 se muestra el diagrama de clases de los monitoreo de los usuarios. Estos diagramas fueron realizados del lado de la capa lógica del proyecto.

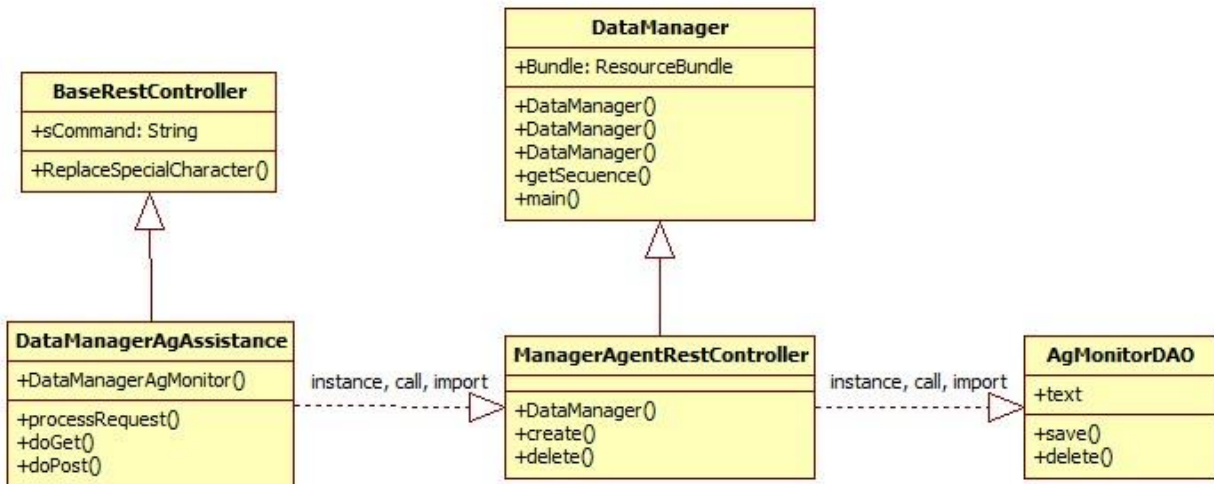


Figura 31. Diagrama de clases del monitoreo de usuarios

8.6.1.3 Modelo de persistencia

El modelo de persistencia permite visualizar la estructura de las tablas de la base de datos y las relaciones de integridad referencial que se modificaron al realizar algunos de los ajustes funcionales planteados en este proyecto. La estructura de estos modelos conserva el patrón definido en la primera fase de Lingweb.

La estructura de estos modelos y de las tablas que los componen, se basaron en previos modelos y tablas de la plataforma Lingweb.

El modelo de persistencia que se presenta en la figura 32, muestra el modelo persistencia para el agente monitor.

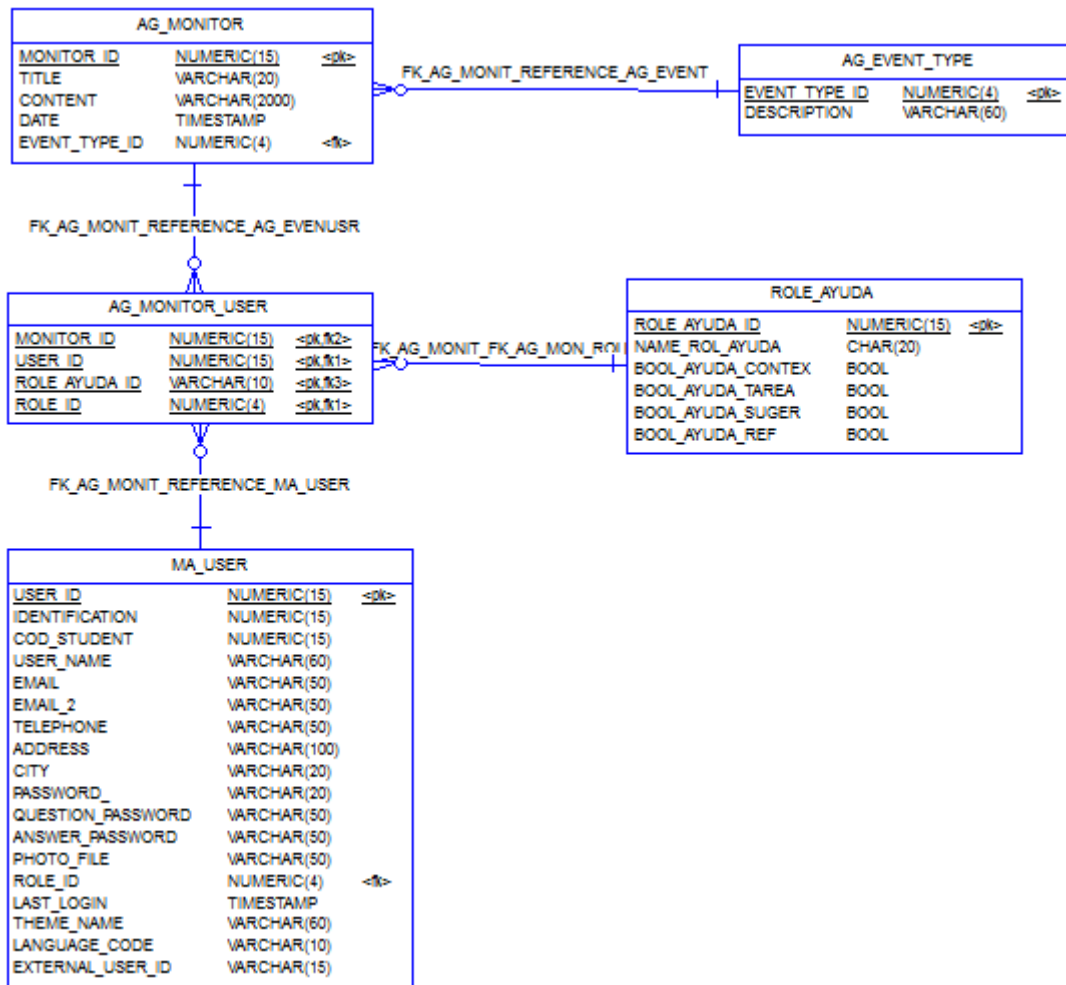


Figura 32. Modelo persistencia del monitoreo de usuarios

Para ver el modelo de persistencia del prototipo remitirse al **ANEXO VI modelo de persistencia**.

8.6 IMPLEMENTACIÓN

Lingweb se encuentran desarrollado bajo un modelo de 3 capas (Modelo, Vista, Controlador) donde la parte de presentación (Interfaz gráfica) es manejada con OpenLaszlo, la parte de lógica y persistencia de datos en Java. La comunicación entre la capa de presentación y lógica se realizar por medio de servicios Web denominados "Servicios REST". Los estándares de creación de los componentes de la figura 29, que comunican cada una de dichas capas, que se detallan a continuación.

8.6.1 CAPA DE PRESENTACIÓN

En esta capa del proyecto, es manejada con la tecnología de Openlaszlo. Una petición de la capa de presentación debe ser creada por medio de la clase <BackendService> como ejemplo, para obtener la lista de cursos, la petición se realizaría de la siguiente manera:

```
<!--Crear asistente de ayuda1-->
<dataset name="createAssistantA1" src="rest/assistance/create" type="http" request="false" proxied="true" />
<BackendService name="createAssistant"
  successDatapath="createAssistantA1:/response/success/message"
```

```

failureDatapath="createAssistantA1:/response/failure/errorMsg"
httpMethod="POST">
<handler name="ontrigger" args="obj">
  if($debug)Debug.write('action ' + this.name + ' triggered');
  this.prepareParams(obj.collectValues());
</handler>
<method name="handleError">
  if($debug)Debug.write('Error creating Assistant!');
</method>
<method name="handleSuccess" args="message,p"><![CDATA[
  if($debug)Debug.write('Assistant created');
  canvas.searchSubnodes("name", "createAssistantForm1").clear();
  this._connector.handleResult(message);
]]></method>
<method name="handleFailure" args="message,type">
  if($debug)Debug.write('Error creating Assistant: '+message);
  this._connector.handleResult(message);
  //si es distinto de validation, es porque la conf ya existe, por tanto se hace una busqueda de la misma
  if(type!="validation") {
    canvas.searchSubnodes("name", "createAssistantForm1").searchAssistantByMet();
  }
</method>
<method name="clearMessage">
  this._connector.handleResult("");
</method>
</BackendService>

```

En el código anterior se puede apreciar que el servicio tiene un dataset encargado de almacenar la respuesta dada por el servidor, esto puede ser cualquier dato con un formato XML. Una vez recibido, Openlaszlo se encarga de procesar los datos basándose en Xpath. En esta clase se indica la dirección del servlet con el que se iría a conectar, junto con el nombre del servicio y el comando a ejecutar.

Cada servicio edita su propio comportamiento después de que la acción haya sido ejecutada. Todos los servicios de la capa de presentación utilizan el servicio <BackendService> para realizar consultas a los servicios REST de la capa lógica.

8.6.2 CAPA LÓGICA

La capa lógica se encuentra representada por los servicios REST y los DataManager o administradores de datos. Los servicios REST son servlets de java que utilizan el protocolo http para recibir peticiones enviadas por la capa de presentación.

Los servicios REST son servlets de Java que reciben las peticiones desde la capa de presentación y las procesan a través del protocolo HTTP. Para la interacción con los agentes se definió el servlet ManagerAgentRestController. En su inicialización se levanta la plataforma Jade y se crea una tabla hash en donde se establece una relación de las acciones que puede recibir y el servlet que las puede realizar. Parte de su codificación se muestra a continuación:

```

public class ManagerAgentRestController extends BaseRestController {
  private static final long serialVersionUID = 1L;
  /**
   * Manejador de mensajes Log's
   */
  protected Logger log = Logger.getLogger(ManagerAgentRestController.class);

```

```

/** Contiene los métodos para la carga de la internacionalización */
protected DataManager dataManager = null;

private Hashtable<String,Action> actions = null;
protected User sender = null;

/**
 * Processes requests for both HTTP <code>GET</code> and <code>POST</code>
 * methods.
 *
 * @param request
 *     servlet request
 * @param response
 *     servlet response
 */
protected void processRequest(HttpServletRequest request, HttpServletResponse response)
    throws ServletException, IOException {

    DOMConfigurator.configure(UserRestController.class.getResource("/log4j.xml"));

    log.info("Parámetros Agentes...");
    printRequestParameter(request);
    // Obtengo el comando del contexto Servlet
    sCommand = this.getRestMethod(request.getRequestURI(), request.getContextPath(), "agent");
    request.setAttribute("command", sCommand);

    // El objeto result usado para pasar los valores de mensajes al JSP
    serviceResult = new RestServiceResult();
    dataManager = new DataManager();

    // Carga el paquete de mensajes para la localización
    dataManager.setBundle(this.loadResourceBundle(request));

    // Respuesta de error por defecto, en caso de que el método REST sea
    // desconocido
    Object[] args = { sCommand };
    serviceResult.setMessage(MessageFormat.format(dataManager.getBundle().getString(
        "rest.unknownMethod"), args));
        log.info("sCommand: " + sCommand);

    //obtenemos la accion a realizar descomente
    String command = request.getParameter("accion"); // nombre de la acción a realizar
    System.out.println("Comando: "+command);

    /**hacemos que el objeto implemente la interfaz accion
    para que pueda utilizar los objetos HttpServletRequest y HttpServletResponse */
    Action action = actions.get(sCommand);

    if (action == null) {
        response.sendError(HttpServletResponse.SC_NOT_IMPLEMENTED);
        log.info("acción nula: "+ action);
        return;
    }
}

```

```

log.info("Ejecutando la acción correspondiente... ");
// TODO HABILITAR CUANDO ESTE EL AGENTE
action.perform(this, request, response);
}

//metodo que se invoca al cargarse el servlet en el contenedor de servlets
public void init() throws ServletException {

    //creación de la tabla hash que contendrá el objeto AccionEnviarMensaje
    actions = new Hashtable<String,Action>();

    //insertamos el objeto con su clave en la tabla hash
    actions.put("createChatClientAgent",new ChatClientAgentAction());
    actions.put("createHelperAgent",new HelperAgentAction());
    actions.put("chatSpeaker",new SendMessageAction());
    actions.put("chatStatus",new ChatStatusAction());
    actions.put("notifySpoken",new NotifySpokenAction());
    actions.put("notifyParticipants",new NotifyParticipantsAction());
    actions.put("annJoinAndLeftUser",new AnnJoinAndLeftUserAction());
    actions.put("listAnnouncementForUser",new AnnouncementUserAction());

    log.info("inicializar el agente");
    log.info("Estado del JadeGateWay :"+JadeGateway.isGatewayActive());

    //Establecemos que clase sera la GateGayAgent
    log.info("Iniciando plataforma...");
    //TODO HABILITAR/DESABILITAR INICIO DE AGENTES
    String args[] = {"-gui", "-port 1099 -container"};
    jade.Boot.main(args);
    JadeGateway.init("edu.univalle.lingweb.agent.agents.GateWayAgent",null);
    log.info("Plataforma Levantada...");
}

    protected void doGet(HttpServletRequest request, HttpServletResponse response) throws ServletException,
        IOException {
        processRequest(request, response);
    }

    protected void doPost(HttpServletRequest request, HttpServletResponse response) throws
ServletException,
        IOException {
        processRequest(request, response);
    }

    /**
     * Returns a short description of the servlet.
     */
    public String getServletInfo() {
        return "Short description";
    }
}

```

Al recibir la petición createHelperAgent, el ManagerAgentRestController hace el llamado al servlet HelperUserAction para que desde él, se envíe la solicitud al agente GateWayAgent (o agente controlador).

Dentro de esta capa, también se incluyen los DataManager encargados de administrar los datos. Para este prototipo se definió DataManagerAgAssistance que se comunica con la capa de persistencia cuando sea necesario. Parte de su código se presenta a continuación:

```
public class DataManagerAgAssistance extends DataManager {

    /**
     * Manejador de mensajes de Log'ss
     * @uml.property name="log"
     * @uml.associationEnd multiplicity="(1 1)"
     */
    private Logger log = Logger.getLogger(DataManagerAgAssistance.class);

    /**
     * Constructor de la clase
     */
    public DataManagerAgAssistance() {
        super();
        DOMConfigurator.configure(DataManagerAgAssistance.class.getResource("/log4j.xml"));
    }

    /**

    */
    public RestServiceResult create(RestServiceResult result, ToAssistance toAssistance) {

        ToAssistanceDAO maAssistanceDAO = new ToAssistanceDAO();

        try {

            toAssistance.setAssistanceId(getSequence("sq_to_assistance"));
            EntityManagerHelper.beginTransaction();
            maAssistanceDAO.save(toAssistance);
            EntityManagerHelper.commit();
            EntityManagerHelper.refresh(toAssistance);

            log.info("Se creó la ayuda con éxito: "+toAssistance.getAssistanceId());
            Object[] args = { toAssistance.getAssistanceId()};

            result.setMessage(MessageFormat.format(bundle.getString("assistance.create.success"), args));

        } catch (PersistenceException e) {
            EntityManagerHelper.rollback();
            e.printStackTrace();
            log.error("Error al guardar la Ayuda: " + e.getMessage());
            result.setError(true);
            result.setMessage(e.getMessage());
        }

        return result;
    }
}
```

8.6.3 CAPA PERSISTENCIA

En esta capa Lingweb tiene representada la información de la base de datos en objetos Java, por medio de JPA. El mapeo de las tablas se realiza por medio del IDE eclipse. Por cada tabla se crean 3 objetos:

- Un objeto interface que permite declarar los métodos que se deben ejecutar para obtener datos de la base de datos.
- Un objeto Abstract que representa la información de campos de la tabla.
- Un Objeto DAO que implementa el objeto “interface” y contiene los métodos de CRUD de la tabla.

Para el caso de mostrar las ayudas, los archivos generados llevan el nombre de AbstractAgAssistence, AgAssistenceDAO y IAgAssistenceDAO. Algunas secciones del código de uno de ellos son:

```
public class AgAssistenceDAO implements IAgAssistenceDAO {
    // property constants
    public static final String TEXT = "text";
    public static final String TEXT_EN = "textEn";
    public static final String TEXT_FR = "textFr";

    private EntityManager getEntityManager() {
        return EntityManagerHelper.getEntityManager();
    }

    public void save(AgAssistence entity) {
        EntityManagerHelper.log("saving AgAssistence instance", Level.INFO, null);
        try {
            getEntityManager().persist(entity);
            EntityManagerHelper.log("save successful", Level.INFO, null);
        } catch (RuntimeException re) {
            EntityManagerHelper.log("save failed", Level.SEVERE, re);
            throw re;
        }
    }

    public void delete(AgAssistence entity) {
        EntityManagerHelper.log("deleting ToAssistence instance", Level.INFO, null);
        try {
            entity = getEntityManager().getReference(AgAssistence.class,
entity.getAssistanceId());

            getEntityManager().remove(entity);
            EntityManagerHelper.log("delete successful", Level.INFO, null);
        } catch (RuntimeException re) {
            EntityManagerHelper.log("delete failed", Level.SEVERE, re);
            throw re;
        }
    }
}
```

La respuesta a cualquier tipo de acceso a la base de datos es redireccionada a la capa lógica. A través de JSP se realiza la generación de la respuesta en XML. Esta respuesta será procesada en la capa de presentación por medio de la clase <dataset>.

9. PRUEBAS

La plataforma Lingweb permite la creación varios tipos de cursos dependiendo de las necesidades del usuario puede ser un curso completo (con sus propias unidades. Secuencias, actividades y ejercicios), cursos por actividades (solo con actividades) y cursos a partir de clonación (curso a partir de un cursos ya existente sea completo o por actividades). El prototipo a creado en este proyecto se centra en el formulario para la creación

de cursos, por lo que las ayudas que crea el sistema solo son para todo lo que tiene que ver solo con este formulario.

Las ayudas que ofrece el prototipo son de dos tipos ayudas invocadas por el usuario, es decir, aquellas que ofrece el sistema a partir de los iconos de ayuda o del link de ayuda; y las ayudas ofrecidas por el sistema como respuesta al monitoreo de las acciones del usuarios como son la ayuda por referencia, la ayuda del comando que es esto, la ayuda sabias que.

Estas pruebas fueron realizadas por la estudiante a cargo del proyecto, bajo un ambiente de trabajo local, debido a que este ambiente no está integrado al sistema Lingweb montado en el servidor de pruebas ni en el servidor de producción.

9.1 PPRUEBAS UNITARIAS

A continuación, en la Tabla 11 se detallan los elementos tenidos en cuenta para la elaboración de la matriz de pruebas unitarias. Además, en la Figura 33 se muestra el formato utilizado para este tipo de pruebas, las cuales se realizaron a los procesos del monitoreo de usuarios .

ELEMENTO DE LA PRUEBA	DESCRIPCIÓN
Módulo	Tipo de secuencia a evaluar.
Código	Código Prueba Identificador de la prueba, con el que se le puede hacer seguimiento
Elemento a probar.	Elemento específico del módulo a probar
Tipo de prueba	Pueden ser dos, de entradas esperadas o inesperadas.
Descripción de la prueba	Descripción de la prueba Descripción de la prueba.
Datos de entrada	Datos necesarios para la ejecución de la prueba.
Acción	Acción que se ejecuta para evaluar la prueba
Datos de salida	Datos de salida Resultado de ejecutar la prueba.
Resultados esperados	¿Se obtuvo el resultado que se esperaba de la prueba?. Se responde sí o no
Resultados esperados ¿Se obtuvo el resultado que se esperaba de la prueba?.	Resultados esperados ¿Se obtuvo el resultado que se esperaba de la prueba?.
Fecha	Fecha en que se ejecutó la prueba
Cumplió (si, no)	¿se clasifica como exitosa la prueba?
Responsable de la prueba	Nombre de la persona responsable de ejecutar la prueba
Responsabilidad de ajustes	Nombre de la persona responsable de realizar los ajustes al elemento probado

Revisiones posteriores: Estos campos sólo serán llenados si el elemento probado no pasó la prueba

Tabla 11. Elementos pruebas unitarias

Como ejemplificación de las pruebas unitarias realizadas, se tiene la prueba con código Mder (Figura 33), que tenía como función enviar el mensaje pidiendo ayuda al agente ayuda ésta no cumplió su objetivo, se pasa corregir el mensaje dado que tiene demasiada información.

Código	Elemento a probar.	Tipo de prueba	Descripción de la prueba	Datos de entrada	Acción	Datos de salida	Resultados esperados	Fecha	Cumplió (si, no)	Observaciones	Responsable de la prueba	Responsabilidad de ajustes
Mder	envio mensaje pidiendo Ayuda para el caso clic derecho en los enlaces	Entradas esperadas	verificar envío mensaje al dar clic derecho en los enlaces de crear curso	no hay entradas	evento clic derecho	mensaje enviado	mensaje al agente ayuda	23/06/2013	si	cambiar el mensaje para el agente ayuda	Luz Adriana Peñaranda G.	Luz Adriana Peñaranda G.
Mtexto	envio mensaje pidiendo Ayuda para el caso clic en los botones del formulario en especial el icono de ayuda de los campos de texto	Entradas esperadas	verificar envío mensaje al dar clic icono ayudadel formulario de crear curso	no hay entradas	evento clic icono ayuda	mensaje enviado	mensaje al agente ayuda	23/06/2013	si	cambiar el mensaje para el agente ayuda	Luz Adriana Peñaranda G.	Luz Adriana Peñaranda G.
MNoEv	Ayuda para el caso de que no se reciban eventos del usuario	Entradas esperadas	verificar envío mensaje al no haber eventos por parte del usuario	no hay entradas	no hay acción	mensaje enviado	mensaje al agente ayuda	23/06/2013	no	no hay envío del mensaje	Luz Adriana Peñaranda G.	Luz Adriana Peñaranda G.

Figura 33. Modelo persistencia del monitoreo de usuarios

9.2 PRUEBAS DE INTEGRACIÓN

A continuación se presenta lo formatos de pruebas utilizados para comprobar el funcionamiento del prototipo.

DOCUMENTOS DE REFERENCIA	
IDENTIFICADOR	TÍTULO
SRS	Sistema de Especificación de Requerimientos

9.2.1 DETALLE DE LA PRUEBA

FECHA REALIZACIÓN:	DURACIÓN:
Requerimientos de la prueba	R 2.4 Monitorear usuarios
Objetivo	Seguir los eventos que representen una acción por parte del usuario. Esto con el fin de ofrecer la información oportuna para el usuario.
Tipo de prueba	Prueba unitaria
Hardware requerido	Equipo donde este instalada la aplicación
Software requerido	No se requiere ningún software
Procedimiento	<ul style="list-style-type: none"> Ejecutar en un navegador la aplicación Ingresar al sistema como usuario profesor (con el perfil básico, intermedio y avanzado) Evaluar los requerimientos a probar

LISTA DE VERIFICACIÓN	
Detalle	Resultados

1. El sistema provee un mecanismo para el monitoreo de los usuarios para que puedan recibir mensajes de ayuda.	Si X	Observaciones
	No	
	Dependiente	
2. Se crea el log que guarda los eventos que representan las acciones del usuario en la aplicación.	Si X	observaciones
	No	
	Dependiente	
3. Se pide las ayudas necesarias según el evento del usuario	Si X	observaciones
	No	
	Dependiente	
<p>Comentarios</p> <p>De acuerdo a los dos tipos de mensajes implementados para el log del usuario en este prototipo se nota que su funcionamiento es coherente. Es necesario proveer de más ayudas al sistema para que el usuario para los demás formularios.</p>		

Tabla 12. Pruebas integración

Los resultados obtenidos en las pruebas funcionales se realizaron para el ambiente de pruebas. Los resultados pueden definirse en tres tipos:

- Si: El requerimiento realizado cumple con lo establecido en el SRS.
- No: El requerimiento no está completamente terminado o funcionalmente correcto. En este caso el requerimiento queda pendiente para una segunda revisión.
- Pendiente: El requerimiento queda aplazado para una posterior prueba o revisión.

Una vez revisados y aprobados los requerimientos establecidos en las pruebas se procedió, a acoplar los ajustes de las funcionalidades existentes y la adición de nuevos requerimientos a la versión que será entregada con este trabajo.

10 CONCLUSIÓN

10. 1 DE LA HERRAMIENTA

- Este trabajo contribuye en el campo de las plataformas virtuales, apoyando a los usuarios a través de la mejora del sistema de ayuda, reduciendo los costos y los tiempos que los estudiantes de estas plataformas invierten en aprender a utilizar las plataformas virtuales, y en buscar asesoría para la utilización de los sistemas. Teniendo en cuenta que el tiempo no es medible en términos económicos, es claro que pensar en proveer ayudas adaptativas, en cierta medida, minimiza la curva de aprendizaje de los usuarios respecto a la interacción con las plataformas, lo cual repercute directamente en el uso, ya que los usuarios tendrán mayor tiempo para realizar tareas más complejas que demanden procesos cognitivos más elaborados.

- El monitoreo de usuarios que propone este proyecto involucra solamente dicho monitoreo en la plataforma Lingweb, esto se hace con el fin de poder ofrecerle al usuario ayudas puntuales dependiendo de la tarea actual que está realizando el usuario y así evitar confundir al usuario con demasiada información.
- El modelo propuesto puede servir de guía para el desarrollo de sistemas de ayudas para otras plataformas e-learning donde el enfoque empleado consista en inferir el conocimiento del usuario mediante la observación de su interacción con el sistema haciendo monitoreo de sus acciones.
- Con este proyecto también se busca motivar a los usuarios a usar sistemas de educación virtual, mejorando la interacción con las ayudas que brindan estos sistemas. El modelo busca brindar ayudas adaptativas que permitan a los usuarios con diferentes nivel de conocimiento hacer un uso eficiente de estos sistemas basándose en su perfil y en la interacción con la aplicación, para adaptar las diferentes ayudas a sus necesidades.

10.2 DE LAS METODOLOGÍAS USADAS

- La metodología RUP Ágil usada para este proyecto, permitió definir las funcionalidades por cada requerimiento descritos en el SRS, lo cual sirvió para tener control y orden sobre lo que se estaba trabajando.
- Trabajar en un ambiente de desarrollo ya definido, hizo que para poder aplicar las arquitecturas y los procesos de finidos en este desarrollo se necesitara del aprendizaje de todas estas metodologías de trabajo y el apoyo de las personas que conforman el proyecto de Lingweb fueran necesarios para la terminación de este proyecto.

10.3 PERSONALES

- Esta fue una experiencia bastante enriquecedora para mi vida tanto personal como profesional; ya que las tecnologías utilizadas eran totalmente desconocidas para mí, además seguir estándares que permitieran desarrollar una aplicación teniendo en cuenta las buenas prácticas en el desarrollo de software. También el hecho de poder aportar a la comunidad académica con la aplicación denominada LingWeb es de gran satisfacción.

11 TRABAJO FUTURO

11. 1 Herramienta para proveer ayudas adaptativas

- Como trabajo futuro se espera la creación de los métodos para la clasificación de los usuarios, en el cual se deben especificar la perfilación de los usuarios del prototipo es decir el usuario básico, intermedio y avanzado, junto con la degradación y la graduación de estos a cada uno de los perfiles nombrados anteriormente.
- También se espera la ampliación de los métodos creados en este prototipo para los usuarios estudiante y administrador.

11. 2 Plataforma Lingweb

- Con respecto a la plataforma Lingweb se espera la implementación de esta modelo, y la integración con esta plataforma en el ambiente de producción.

12 REFERENCIAS

- [Aricapa y Puentes 2009] Aricapa J. y Puentes J. "Ambiente virtual basado en una arquitectura multiagente para sistemas e-learning centrados en la enseñanza de idiomas (se-mas)", Tesis Pregrado, Escuela de Ingeniería de Sistemas y Computación, Universidad del Valle, Santiago de Cali, Colombia, 2009.
- [Bedoya 2003] Bedoya O. Inteligencia Artificial, IESC Universidad del Valle. Imagen adaptada del libro "Inteligencia Artificial. Un Enfoque Moderno". 2a. Edición. Pearson Prentice Hall, Peter Norvig y Stuart Russell. 2003. Visitado noviembre 2011. Disponible en: <http://eisc.univalle.edu.co/~oscarbed/IA/Tema2.pdf>
- [Boneu 2007] Boneu J. M. Plataformas abiertas de e-learning para el soporte de contenidos educativos abiertos. Revista de universidad y sociedad del conocimiento, 2007: 36-47
- [Breuker 1990] Breuker, J. 3. (Eds), "Developing Intelligent Help Systems. Report on the P280 ESPRIT Project EUROHELP. EC, Kopenhagen. 1990.
- [Brusilovsky 1996] Brusilovsky, P. (1996). Methods and techniques of adaptive hypermedia. User Modeling User-Adapted Interaction '96, v 6, pág: 87-129.
- [Brusilovsky, Schwarz & Weber 1996] Brusilovsky P., Schwarz E., and Weber G. 1996, .ELM-ART: An intelligent tutoring system on World Wide Web. Third International Conference on Intelligent Tutoring Systems, ITS-96, Montreal.
- [Chin 1989] Chin, D. N. User Models in Dialog System, capítulo KNOWME: Modeling What the User Knows in UC, paginas 74-107. Springer, Berling, 1989.
- [Deagostini & Cormenzana 2007] Deagostini Adriana, Cormenzana Fernando. (visitado 2011, noviembre). Interfaces de Usuario Inteligentes: Sistemas Adaptativos, Curso: Interacción humano computador y diseño de interfaces, Prof: Dr. Rosario Girardi. Disponible en: <http://www.crnti.edu.uy/05trabajos/interface/InterfacesdeUsuarioInteligentes.doc>
- [Eclipse] Sitio Oficial de Eclipse. (visitado 2011, noviembre). Disponible en <http://eclipse.org/>
- [Eclipse 2] Proyecto JPA en Eclipse. (visitado 2011, noviembre). Disponible en <http://www.eclipse.org/eclipselink/>
- [Fernández 2001] Fernández M. B., "Sistemas de ayuda inteligente para entornos informáticos complejos". Inteligencia Artificial, Revista Iberoamericana de Inteligencia Artificial. No.12, pp. 59-67, 2001. [Online]. Artículo disponible: <http://www.fdi.ucm.es/profesor/balta/publicaciones/RIIAbalta.pdf>
- [Fenstermacher y Ginsburng 2002] Fenstermacher Kurt D, Ginsburng Mark. "A Lightweight Framework for Cross-Application User Monitoring". IEEE. Marzo 2002.
- [Galitz 2007] Galitz W. O., "The essential Guide to User Interface Desing: An Introduction to GUI Principles and Techniques", 3ra edición, Wiley Publishing Inc, pp: 606-623. 2007.
- [Guerra 1997] Guerra Hernández A., Agentes Interfaz Inteligentes, Maestría en Inteligencia Artificial, Universidad Veracruzana - LANIA, A.C. Sebastián Camacho No. 5, Xalapa, Ver. 91000
- [Höök 1996] Höök, K. A Glass Box Approach to Adaptive Hypermedia. PhD tesis, Stockholm University, Swedish Institute of Computer Science, 1996.
- [Iglesakis 2004] Iglesias D. Adaptive Help Web based Applications. Adaptive Hypermedia and Adaptive Web Based System. Third International Conference. 2004, pp. 304-307, 2010.
- [INGENIAS 2005] "INGENIAS". Universidad Complutense de Madrid (GRACIA). 2005, [online]. Disponible en:

<http://grasia.fdi.ucm.es/ingenias>.

[JADE] Sitio oficial de JADE. (visitado 2011, noviembre) Disponible en <http://jade.tilab.com>

[Jimenez 2000] Jimenez Silvestre, Ramos Esmeralda, Agentes Inteligentes, Universidad Central de Venezuela, Caracas, Venezuela, 2000, ISSN: 1316-6239.

[Kearsley 1988] Kearsley, G. Online Help System: "Desing and Implementation," Ablex Publishing Corporation, Norwood, New Jersey, USA, 1988.

[Kuldeep y Poonphon 2008] Kuldeep Nagi, Poonphon Suesawaluk. "Research Analysis of Moodle Reports to Gauge the Level of Interactivity in eLearning Courses at Assumption University, Thailand". International Conference on Computer and Communication Engineering 2008. IEEE 2008

[Luy, Kitagatay, Suganumay, & Kinoshita 2003] Luy, L., Kitagatay, G., Suganumay, T., & Kinoshita, T. (2003). Adaptive User Interface for Multimedia Communication System Based on Multiagent. IEEE Proceedings of the 17th International Conference on Advanced InformationNetworking and Applications.

[Log4j] Sitio oficial log4j. (visitado 2011, noviembre). Disponible en <http://logging.apache.org/log4j/1.2>

[Machuca 2009] Machuca V. Liliana E. "Arquitectura multiagente para un sistema e-learning centrado en la enseñanza de idiomas (se-mas)". Tesis MsC, Escuela de Ingeniería en Sistemas y Computación Universidad del Valle, Santiago de Cali, Colombia, 2009.

[Machuca y Rodríguez 2011] Machuca V. Liliana E., Rodríguez C. Paola J. "Experiences in developing an e-learning system for language learning and teaching". En IEEE Computing Congress (CCC), 2011 6th Colombia, pp. 1-6. Colombia 2011.

[Mas 2005] Mas A., Agentes Software y Sistemas Multi-Agente: Conceptos, Arquitecturas y Aplicaciones. Edición Pearson Prentice Hall, Madrid, 2005. ISBN 84-205-4367-5.

[Molina2004] Molina López José, García Herrero Jesús, Bernardos Barbolla Ana M^a, Agentes y Sistemas Multiagente, Universidad carlos III y universidad Politécnica, Madrid. Disponible en: <http://www.ceditec.etsit.upm.es/index.php/Descargar-documento/3-Agentesy-Sistemas-Multiagente.html>

[Openlaszlo 1] Sitio Oficial de OpenLaszlo. (visitado 2011, noviembre) Disponible en <http://www.openlaszlo.org/>

[Openlaszlo 2] Sitio Web en Español. (visitado 2011, noviembre) Disponible en <http://www.openlaszlo.net/>

[Oyama, Takeushi, Ming y Chang 2011] Oyama Katsunori, Takeushi Atsushi, Ming Hua, Chang Carl K.. "A Concept Lattice for Recognition of User Problems in Real User Monitoring". 18th Asia-Pacific Software Engineering Conference. IEEE 2011

[Peña 2004] Peña Clara Inés, Intelligents Agents to improve adaptativity in a Web based learning environment, Universidad de Girona, España 2004. ISBN: 84-688-6950-3.

[Padgham & Winikoff 2004] Padgham Lin & Winikoff Michael. Developing Intelligent Agent Systems: A practical guide. John Wiley & Sons Ltd, England, 2004. ISBN 0-470-86120-7 (HB)

[Pikoulas, Buchanan y Triantafyllopoulos 2002] Pikoulas J., W. Buchanan, Mannion M., Triantafyllopoulos K. "An Intelligent Agent Security Intrusion System". Ninth Annual IEEE International Conference Workshop on the Engineering of Computer-Based System (ECBS'02). IEEE 2002.

[Postgres] Sitio Oficial de PostgreSQL. (visitado 2011, noviembre) Disponible en <http://www.postgresql.org/>

[Raza 2009] Raza Abidi, Syed Sibte. (2009). Intelligent Information Personalization: From Issues to Strategies. En C. Mourlas, & P. Germanakos, Intelligent User Interfaces: Adaptation and Personalization Systems and Technologies (págs. 118-142). Hershey - New York: Informatlon science reference.

[Rodríguez y Machuca 2009] Rodríguez C. Paola J., Machuca V. Liliana E. "Arquitectura multiagente para un sistema e-learning centrado en la enseñanza de idiomas". En Sánchez (Ed): Nuevas Ideas en Informática Educativa. Volumen 5, pp 83-91. 2009-2011.

- [Sagiroglu y Canbek 2009]** Sagiroglu Seref, Canbek Gurol. "Keylogger: Increasing Threats to Computer Security and Privacy". IEEE Technology and Society Magazine, Volume 28. IEEE 2009
- [Santacruz V. Liliana P. 2008]** Santacruz V. Liliana P. "agentes taxonomía y aplicaciones", Universidad Carlos III de Madrid, Visitado noviembre 2011, disponible en: <http://www.lite.etsii.urjc.es/liliana/agentes.pdf>
- [spket]** Plug-in para editor OpenLaszlo en Eclipse. Incluye versión Eclipse Europa 3.3 utilizada en LingWeb. (visitado 2011, noviembre) Disponible en <http://tiny.spket.com/>
- [Subversion]** Sitio Oficial de Subversión (SVN). Incluye plug-in para Eclipse. (visitado 2011, noviembre) Disponible en <http://subversion.tigris.org/>
- [Uchyigit 2009]** Uchyigit, Gulden. The Next Generation of Personalization Techniques. En Intelligent User Interfaces: Adaptation and Personalization Systems and Technologies , de Constantinos Mourlas y Panagiotis Germanakos, 72-87. Hershey - New York: Information science reference, 2009.
- [Velásquez Y Palade, 2008]** Velásquez, J. D., & Palade, B. (2008). Adaptive Web Sites: A Knowledge Extraction from Web Data Approach. Proceedings Frontiers in Artificial Intelligence and Application Volume 170, IOS Press.
- [Winkels 1992]** Winkels, R., Explorations in Intelligent Tutoring and Help. IOS Press, Amsterdam. 1992.