



Ingeniería y Competitividad

ISSN: 0123-3033

inycompe@gmail.com

Universidad del Valle

Colombia

Rivera, Diana C.; Machuca-Villegas, Liliana E.

Arquitectura de un asistente virtual para la producción de textos en la enseñanza y aprendizaje de
idiomas

Ingeniería y Competitividad, vol. 16, núm. 2, 2014, pp. 155-164

Universidad del Valle

Cali, Colombia

Disponible en: <http://www.redalyc.org/articulo.oa?id=291333276014>

- Cómo citar el artículo
- Número completo
- Más información del artículo
- Página de la revista en redalyc.org

redalyc.org

Sistema de Información Científica

Red de Revistas Científicas de América Latina, el Caribe, España y Portugal
Proyecto académico sin fines de lucro, desarrollado bajo la iniciativa de acceso abierto

Arquitectura de un asistente virtual para la producción de textos en la enseñanza y aprendizaje de idiomas

SYSTEMS ENGINEERING

A virtual assistant architecture to support the delivery of texts in languages teaching - learning processes

Diana C. Rivera*, Liliana E. Machuca-Villegas **§

*Universidad Cooperativa de Colombia, Cali, Colombia, **§Escuela de Ingeniería de Sistemas,
Facultad de Ingeniería, Universidad del Valle, Cali, Colombia
diana.rivera@campusucc.edu.co, liliana.machuca@correounalvalle.edu.co

(Recibido:24 de Abril de 2013-Aceptado: 09 de Junio de 2014)

Resumen

Este artículo presenta la arquitectura de un Asistente virtual que ofrece funciones de apoyo al usuario en el proceso de producción de textos para la enseñanza y el aprendizaje de idiomas. Se definen dos funciones específicas para el comportamiento del asistente: la asistencia de escritura y la asistencia gramatical. En la asistencia de escritura se le presenta al estudiante recomendaciones de acuerdo al tipo de párrafo que desea escribir, y en la asistencia gramatical se le brinda al estudiante la corrección ortográfica y sintáctica de oraciones de los párrafos o textos que produce. El asistente integra características de agentes de software, ambientes virtuales de aprendizaje y elementos de la lingüística computacional que permiten realizar el procesamiento del lenguaje y dar recomendaciones a los estudiantes sobre su proceso de escritura. El principal propósito de este asistente es ayudar a los estudiantes a mejorar sus habilidades de escritura cuando interactúen en un entorno de aprendizaje virtual. La arquitectura propuesta se ha implementado en la plataforma denominada Lingweb, prototipo de un Sistema de Gestión de Cursos para la enseñanza y aprendizaje de idiomas.

Palabras clave: Agentes de software asistente virtual, entorno virtual de aprendizaje, producción textual.

Abstract

This paper presents the architecture of a Virtual Assistant that supports the delivery of texts in languages teaching-learning processes. Two specific functions for the assistant are defined: writing and grammatical assistance. On one hand, writing assistance provides students recommendations about the type of paragraph that they want to write. And, on the other hand, grammatical assistance provides students a feedback about their mistakes in the texts that they produce. The assistant has features related with three technological elements: i) software agents, ii) virtual learning environments and iii) computational linguistics. These features allow to provide the functionalities for language processing and they provide students' recommendations about their writing processes. The main purpose of this assistant is to help the students to improve their writing skills, when they interact into a virtual learning environment. The proposed architecture was implemented in a platform called Lingweb, which is a prototype of a Course Management System for the languages teaching-learning processes.

Keywords: Software Agents, textual Production, virtual Assistant, virtual learning environments.

1. Introducción

Uno de los resultados significativos que se han obtenido en el marco del proyecto de investigación “Ambiente virtual para el aprendizaje de idiomas” (proyecto cofinanciado entre la Universidad del Valle y Colciencias) es la plataforma Lingweb, una plataforma virtual basada en agentes software (Machuca y Rodríguez, 2011). Esta plataforma se desarrolla como una instancia de medicación pedagógica para los procesos de enseñanza y aprendizaje de los idiomas.

Lingweb además de tener las características de un sistema de gestión de cursos en línea (CMS, por sus siglas en inglés) cuenta con herramientas y ejercicios que brindan apoyo en el desarrollo de las habilidades de comunicación como la lectura, escritura, escucha y conversación, habilidades requeridas en el aprendizaje de cualquier idioma. En la primera versión de Lingweb se hizo especial énfasis en el acompañamiento del proceso de escritura, desarrollando herramientas de procesamiento lingüístico como el editor avanzado que incluye las opciones de revisión de textos y marcado de errores. También se desarrollaron ejercicios para estudiantes basados en plantillas para la escritura de párrafos y textos (Berdugo, Herrera y Valdiri, 2010).

Aunque existen muchos Sistemas de Gestión de Aprendizaje (LMS, por su sigla en inglés) tales como Blackboard.com, Moodle.org, Sakaiproject.org, Dotlrn.org entre otros, los cuales asisten procesos de enseñanza aprendizaje tanto en el contexto educativo como en el corporativo; este tipo de sistemas presentan algunas falencias a la hora de asistir a los estudiantes en los procesos de enseñanza-aprendizaje. Específicamente se puede mencionar las limitaciones de las funcionalidades relacionadas con el apoyo en el proceso de escritura que un estudiante puede requerir. Por otra parte, para el desarrollo de la destreza en la escritura en este tipo de ambientes virtuales de aprendizaje hace falta que el estudiante tenga experiencias de producción textual, en las cuales cuente con cierta asistencia en el proceso de elaboración y redacción, de tal modo que se tenga un acompañamiento constante

y una retroalimentación apropiada de su desempeño lingüístico (Berdugo, Herrera y Valdiri, 2010).

Teniendo en cuenta las necesidades mencionadas anteriormente y tomando como base las herramientas de escritura con las que cuenta Lingweb, se desarrolló un Asistente virtual integrado a la plataforma, que se puede catalogar como un Sistema de Asistencia a la Enseñanza (TAS, por sus siglas en inglés) ya que realiza funciones de apoyo al estudiante en el proceso de producción de textos, generando recomendaciones a nivel léxico, sintáctico, de escritura y macrotextual. La idea es que un usuario pueda detectar las falencias en sus composiciones textuales, antes de ser enviado al profesor para su respectiva calificación (Rivera, 2011).

Para la implementación del asistente virtual se tomaron características de tres elementos tecnológicos a saber: agentes software, procesamiento de lenguaje natural y ambientes virtuales de aprendizaje. Para tal fin se hizo una revisión de literatura en donde se encontraron varios trabajos basados en agentes software aplicados a sistemas conversacionales y desarrollos en el procesamiento del lenguaje natural como, Ross *et al.* (2004); Traum & Rickel, (2002); Sidner *et al.* (2000); McRoy *et al.* (2001), por citar algunos. Dentro de estos trabajos no se halló un proyecto que utilizara conjuntamente estos elementos tecnológicos (Rivera, 2011). En este sentido, se definió una arquitectura que reunió características de un agente de software (Mas 2005) (Peña, 2002) y elementos de la lingüística computacional que permitieron realizar el procesamiento del lenguaje y dar recomendaciones a los estudiantes sobre su proceso de escritura en un ambiente de aprendizaje virtual.

También se tomaron características del modelo pedagógico de la plataforma virtual Lingweb, el cual está basado en la concepción socioconstructivista de la enseñanza y el aprendizaje (Berdugo y Pedraza, 2010). El modelo busca ofrecer un diseño que promueva procesos de construcción de conocimiento, en particular, del conocimiento lingüístico y de las competencias comunicativas en las lenguas extranjeras, desde los planos individual y social. Se complementa con los enfoques teóricos sociocultural, interaccionista,

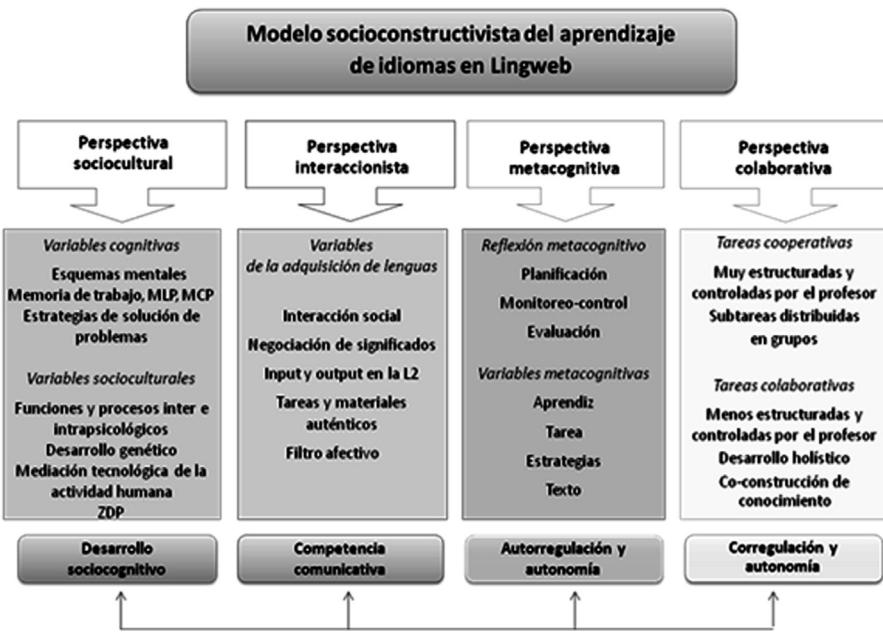


Figura 1. *Modelo pedagógico de Lingweb (Berdugo y Pedraza, 2010)*

metacognitivo y colaborativo del aprendizaje de lenguas. La representación de los cuatro principios mencionados se muestra en la Figura 1, en ella se resumen las especificaciones de cada uno. Se aprecian características cognitivas, socioculturales, la interacción en un plano real, la conciencia que se tiene del conocimiento, la colaboración y la cooperación (Berdugo y Pedraza, 2010). En cuanto a la producción textual para la enseñanza de un idioma, el modelo pedagógico se enfoca en la composición escrita por procesos, identificando tres momentos: la fase de preescritura (planeación), la fase de escritura propiamente (textualización) y finalmente la fase de revisión (Berdugo y Pedraza, 2010), estos momentos son configurados desde el diseño de los cursos en cada una de sus secuencias de aprendizaje.

2. Metodología

Este trabajo fue de carácter interdisciplinario. Integró las áreas de Ingeniería de Software Orientada a Agentes (AOSE), ambientes virtuales de aprendizaje, procesamiento del lenguaje natural y lingüística aplicada a la enseñanza de lenguas. Adoptó los procedimientos metodológicos de dos enfoques investigativos que se integraron durante el proceso de desarrollo: en primer lugar, la Ingeniería de Software Orientada a Agentes con la metodología Ingenias (Gómez, 2002; Pavón y Gómez, 2003), que orientó el diseño e implementación del asistente; y en

segundo lugar, la Investigación-Acción (I-A), que orientó el desarrollo de las necesidades del asistente desde un contexto educativo, contando con la participación y el acompañamiento de las profesoras de La Escuela de Ciencias del Lenguaje líderes del proyecto Lingweb. Se desarrollaron actividades enfocadas en la recolección de información, en la definición de las necesidades pedagógicas y tecnológicas, en el diseño e implementación de la arquitectura y en la evaluación de las funcionalidades del asistente. En cada una de estas actividades se tuvo en cuenta los principios para la enseñanza y el aprendizaje de lenguas definidos en el modelo pedagógico de Lingweb.

A continuación se describen las actividades que se siguieron para el desarrollo de la arquitectura del asistente.

2.1 Análisis y diseño de necesidades educativas

En esta actividad se establecieron los requerimientos del asistente desde un contexto tecnológico y pedagógico. Se definieron los usuarios del software, identificando como usuarios a profesores y estudiantes de la plataforma de aprendizaje Lingweb de la Escuela de Ciencias del Lenguaje de la Universidad del Valle. Durante esta etapa, se realizaron continuas sesiones de trabajo con las

profesoras de idiomas, quienes ayudaron a definir las necesidades educativas en función de la estructura y características que se debían tener en cuenta para realizar una composición textual, así como la corrección ortográfica y sintáctica de oraciones.

De esta actividad se obtuvo el documento ERS (Especificación de Requisitos Software) el cual siguió la norma IEEE 830 para su desarrollo. Las funcionalidades definidas para el asistente a un nivel general son: realizar la revisión ortográfica de las oraciones; proporcionar el análisis sintáctico a los textos escritos; presentar sugerencias en cuanto a estructura y características de los tipos de textos expositivos y argumentativos (reseña y ensayos); presentar sugerencias de acuerdo a las características de los tipos de párrafo expositivo y argumentativo; y proporcionar una herramienta para el marcado de errores.

2.2 Análisis y definición de los lexicones y analizadores sintácticos

Se estudiaron varios analizadores léxicos y sintácticos con el fin de evaluar su compatibilidad con Lingweb y con las funcionalidades del asistente. Los analizadores debían funcionar en ambientes web, ser de libre distribución y aplicar el procesamiento de lenguaje natural a la escritura de textos. En la Tabla 1. se presentan algunos de los resultados obtenidos del estudio de los analizadores explorados. Debido a que ninguno de los analizadores estudiados cumplió con las características necesarias para realizar el asistente virtual, se optó por seleccionar algunos elementos de estos analizadores y desarrollar nuevos componentes de acuerdo a lo requerido para el asistente con relación al lexicón, analizador ortográfico y analizador sintáctico.

2.3 Diseño de la arquitectura del asistente virtual

2.3.1 Vista conceptual de la arquitectura del asistente

La arquitectura del asistente se diseñó de acuerdo con las especificaciones arquitecturales y técnicas de LingWeb (Aricapa y Puentes, 2009; Machuca, 2010). LingWeb es una aplicación que se basa en una arquitectura de tres capas compatible con

JADE (Java Agent Development FrameWork) (Jade, 2012) para permitir el trabajo con agentes software. En el lado del servidor se trabaja con J2EE por su confiabilidad, robustez, y compatibilidad con la plataforma JADE; del lado del cliente se optó por la plataforma OpenLaszlo (Laszlo, 2012), la cual ofrece características RIA ideal para el uso en las aplicaciones Web actuales.

Conceptualmente Lingweb está estructurado por secciones o módulos que hacen referencia a la parte de administración, gestión y diseño de cursos, herramientas de estudio y comunicación, adaptatividad e interfaz gráfica. Es en el módulo de gestión y diseño de cursos donde se integra el asistente con Lingweb y se ofrece la asistencia de escritura y asistencia gramatical, la cual se hace presente en el momento de escribir un texto. La asistencia cuenta con un editor de texto especializado llamado Editor LingWeb que ha sido adaptado del editor por defecto de Openlaszlo. Este editor pone a disposición de docentes novedosas herramientas como el marcado de errores, herramienta clásica inspirada en el proceso de corrección en papel creado por Holmes (2012). Con el marcado de errores se permite identificar aquellas fallas de escritura relacionadas con la gramática, la sintaxis, la cohesión, la puntuación y el vocabulario.

Como se mencionó anteriormente, el Asistente virtual tiene dos funciones específicas que lo definen: la asistencia de escritura y la asistencia gramatical. En la asistencia de escritura se le presenta al estudiante recomendaciones de acuerdo con el tipo de párrafo que desea escribir (recomendaciones macrotextuales) y en la asistencia gramatical se le brinda al estudiante la corrección ortográfica y sintáctica de oraciones de los párrafos o textos que produce. Estas funciones son implementadas por dos agentes de software como se aprecia en la Figura 2.

2.3.2 Vista lógica de la arquitectura del asistente

La arquitectura del asistente está constituida por tres capas: la capa de presentación, la capa lógica y la capa de almacenamiento. En la capa de presentación se encuentran las funcionalidades de

Tabla 1. Resultados evaluación analizadores léxico y sintáctico (Rivera, 2011)

NOMBRE DEL ANALIZADOR		CRITERIOS DE SELECCIÓN							
		INTEGRACION CON JAVA	LICENCIA DE SW LIBRE	CONTEXTO DE APLICACIÓN GENERAL	IDIOMA INGLES	CODIGO ABIERTO	IMPLEMENTACION EN TECNOLOGÍA WEB	ALGORITMO EFICIENTE	EL LEXICON HACE USO DE BD.
		NO	SI	NO	NO	SI	NO	NO	NO
<i>OBSERVACIONES</i>									
<i>DYPAR -I</i> Este intérprete no es un sistema para analizar (parser) textos en lenguaje natural de cualquier dominio y sin restricciones. Está construido solo para su aplicación en el mundo de los bloques, además el lenguaje sobre el que esta implementado no es de fácil integración con la estructura base del proyecto.									
		INTEGRACION CON JAVA	LICENCIA DE SW LIBRE	CONTEXTO DE APLICACIÓN GENERAL	IDIOMA INGLES	CODIGO ABIERTO	IMPLEMENTACION EN TECNOLOGÍA WEB	ALGORITMO EFICIENTE	EL LEXICON HACE USO DE BD.
		NO	NO	SI	SI	NO	SI	NO	SI
<i>OBSERVACIONES</i>									
<i>ASPSpellCheck</i> Es un corrector ortográfico diseñado solo para trabajar con software desarrollado por Microsoft, así que no hay la disponibilidad de conocer cómo está elaborado no se adaptó porque el proyecto se trabaja bajo licencia de libre distribución.									
		INTEGRACION CON JAVA	LICENCIA DE SW LIBRE	CONTEXTO DE APLICACIÓN GENERAL	IDIOMA INGLES	CODIGO ABIERTO	IMPLEMENTACION EN TECNOLOGÍA WEB	ALGORITMO EFICIENTE	EL LEXICON HACE USO DE BD.
		SI	NO	SI	SI	NO	SI	NO	SI
<i>OBSERVACIONES</i>									
<i>Java Script SpellChecker</i> Es un corrector ortográfico diseñado solo para trabajar con software desarrollado por Microsoft, no se empleó para el proyecto porque no hay la disponibilidad de conocer cómo está elaborada la herramienta o el algoritmo utilizado para la corrección de errores. Además, el alojamiento del lexicón se hace en una base de datos que utiliza servidores Windows, por lo tanto también debe existir algún tipo de licenciamiento para utilizar estas aplicaciones.									
		INTEGRACION CON JAVA	LICENCIA DE SW LIBRE	CONTEXTO DE APLICACIÓN GENERAL	IDIOMA INGLES	CODIGO ABIERTO	IMPLEMENTACION EN TECNOLOGÍA WEB	ALGORITMO EFICIENTE	EL LEXICON HACE USO DE BD.
		SI	SI	SI	SI	NO	SI	NO	SI
<i>OBSERVACIONES</i>									
<i>Word Net</i> WordNet es un sistema electrónico de referencia léxica, desarrollado en forma de base de datos léxica. La diferencia básica entre éste y otros proyectos es que es el único proyecto que relativamente a gran escala se ha tenido como idea fundamental la organización del léxico en campos semánticos, además de que gracias a las posibilidades que ofrece tiene una gran cantidad de información redundante entre ellas se tiene: sinonimia, antonimia, hiponimia/hiperonimia, holonimia/meronimia y morfológico. No se utiliza para el proyecto porque realiza un tipo de análisis diferente a lo requerido.									
		INTEGRACION CON JAVA	LICENCIA DE SW LIBRE	CONTEXTO DE APLICACIÓN GENERAL	IDIOMA INGLES	CODIGO ABIERTO	IMPLEMENTACION EN TECNOLOGÍA WEB	ALGORITMO EFICIENTE	EL LEXICON HACE USO DE BD.
		SI	SI	SI	SI	NO	SI	SI	SI
<i>OBSERVACIONES</i>									
<i>OpenNLP</i> Es un proyecto referente al PLN desarrollado en Java. Su código es libre y entre sus utilidades destacan la detección de sentencias, tokenización de textos. Dichas utilidades están disponibles para procesar textos en inglés y castellano. No se utilizó en el proyecto porque provee muchas más utilidades de las necesarias.									
		INTEGRACION CON JAVA	LICENCIA DE SW LIBRE	CONTEXTO DE APLICACIÓN GENERAL	IDIOMA INGLES	CODIGO ABIERTO	IMPLEMENTACION EN TECNOLOGÍA WEB	ALGORITMO EFICIENTE	EL LEXICON HACE USO DE BD.
		SI	SI	SI	SI	NO	SI	SI	SI
<i>OBSERVACIONES</i>									
<i>Freeling</i> Es una herramienta de libre distribución implementada en C++. Está diseñada para ser utilizada como una librería externa desde una aplicación, y entre las utilidades que se extraen de esta herramienta destacan: Tokenización de texto, separación de sentencias, análisis morfológico, reconocimiento flexible de multipalabras y detección y clasificación de entidades nombradas. A esto se añade que está basada en diccionarios. No se utilizó porque está desarrollada en un lenguaje de programación diferente al requerido.									
		INTEGRACION CON JAVA	LICENCIA DE SW LIBRE	CONTEXTO DE APLICACIÓN GENERAL	IDIOMA INGLES	CODIGO ABIERTO	IMPLEMENTACION EN TECNOLOGÍA WEB	ALGORITMO EFICIENTE	EL LEXICON HACE USO DE BD.
		NO	SI	SI	SI	NO	SI	---	NO
<i>OBSERVACIONES</i>									
<i>Link Grammar</i> Es un analizador sintáctico de inglés, basado en la gramática de enlace, una teoría original de la sintaxis Inglés. No se utilizó porque el código no se encuentra disponible y no se puede manipular.									
		INTEGRACION CON JAVA	LICENCIA DE SW LIBRE	CONTEXTO DE APLICACIÓN GENERAL	IDIOMA INGLES	CODIGO ABIERTO	IMPLEMENTACION EN TECNOLOGÍA WEB	ALGORITMO EFICIENTE	EL LEXICON HACE USO DE BD.
		SI	SI	SI	SI	NO	NO	---	NO
<i>OBSERVACIONES</i>									

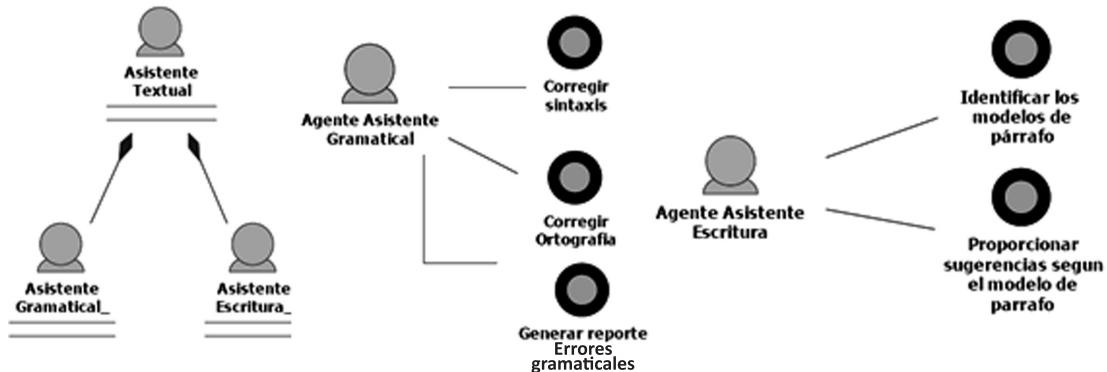


Figura 2. Modelo de agentes asistente textual

configuración, elaboración y revisión de ejercicios de escritura los cuales están conformados por ejercicios tipo texto y ejercicios tipo párrafo. Para dichos ejercicios se diseñaron plantillas de escritura que se basan en el modelo pedagógico propuesto por Berdugo y Pedraza (2010). Por medio de las plantillas se establece el tipo de párrafo a crear, el número de secciones por las que está compuesto, la descripción de cada sección y las recomendaciones que el asistente debe realizar para cada una de éstas. En la Figura 3 se presenta un ejemplo de una plantilla tipo párrafo en conjunto con el editor Lingweb.

La capa lógica de la aplicación contempla todo lo relacionado con la asistencia textual que se le proporciona al usuario, está representada por medio del asistente grammatical y del asistente de escritura. Para la capa de almacenamiento se utiliza la base de datos Lingweb la cual dentro de sus componentes especializados para este proyecto cuenta con un lexicón que sirve de ayuda para la realización de corrección ortográfica; además, contiene los recursos sintácticos que almacenan las reglas sintácticas comunes para el idioma inglés y los modelos de párrafos, identificando su clasificación y secciones asociadas a cada uno de ellos.

De acuerdo a la estructura conceptual del sistema y a su arquitectura por capas, la arquitectura en Lingweb para el asistente virtual desde su función de escritura queda establecida como lo refleja la Figura 4 y desde su función grammatical en la Figura 5. Estas figuras muestran cada uno de los componentes de los asistentes y la función que desempeñan.

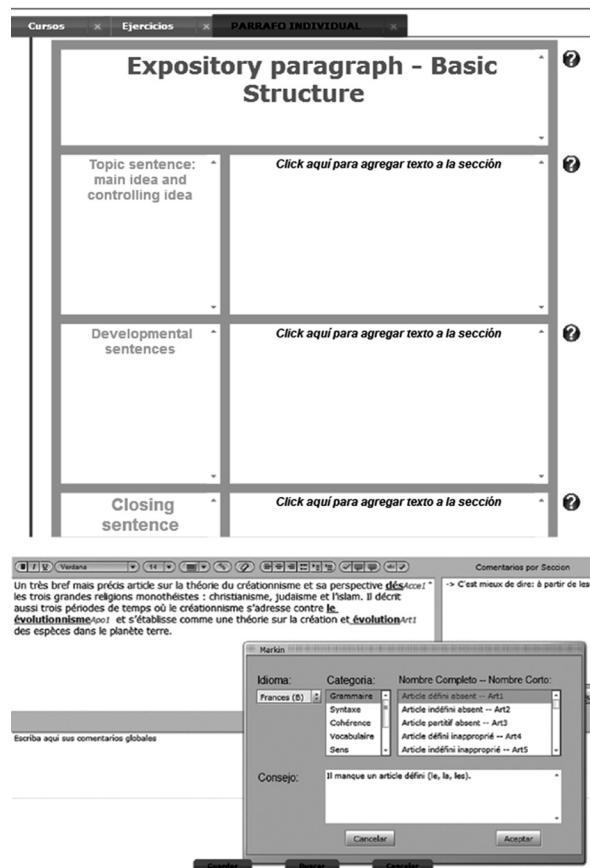


Figura 3. Interfaz plantilla tipo párrafo y Editor Lingweb.

2.4 Implementación del asistente

La arquitectura del Asistente virtual se implementó como un módulo de la plataforma Lingweb. Se utilizaron tecnologías de acuerdo a las diferentes capas del sistema, buscando mantener siempre la arquitectura establecida para el proyecto. La

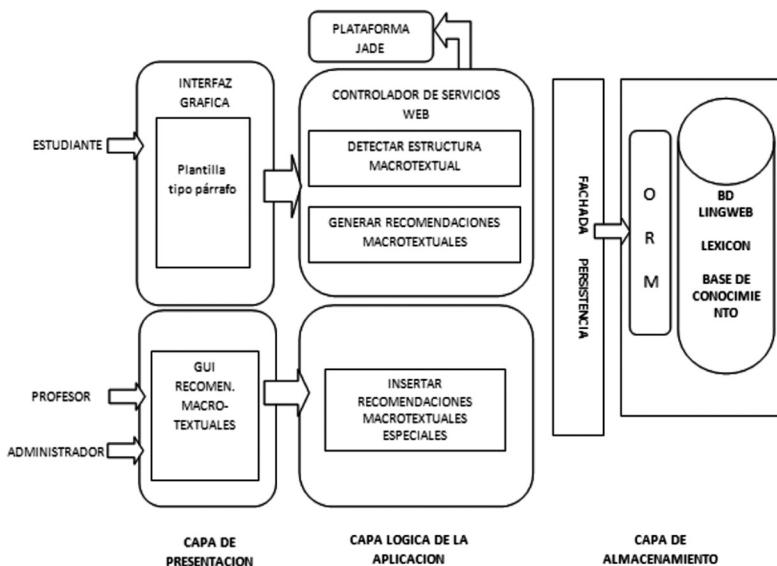


Figura 4. Estructura general de la arquitectura del asistente de escritura

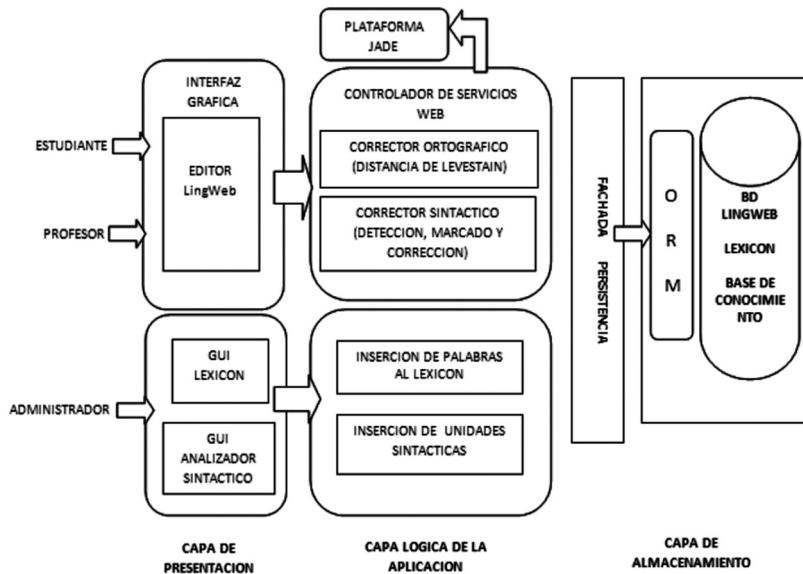


Figura 5. Estructura general de la arquitectura del asistente gramatical

descripción de la integración de las tecnologías con las capas de la arquitectura son descritas en (Aricapa y Puentes, 2009; Machuca, 2010; Machuca y Rodríguez, 2011).

Las recomendaciones que se generan a nivel gramatical y de escritura se implementan por medio de las herramientas de corrección ortográfica y sintáctica que están disponibles en el editor Lingweb y por medio del monitoreo de las plantillas de escritura para ofrecer su ayuda de acuerdo al tipo de párrafo que se desea redactar. Por medio del monitoreo, el

asistente detecta si existen errores de cualquier tipo, si es así el asistente genera un mensaje presentando el error. La revisión se realiza en una ventana independiente la cual presenta el texto indicando las fallas y las sugerencias correspondientes.

Desde el perfil del profesor se puede configurar el tipo de párrafo que debe desarrollar el estudiante, para ello se selecciona la plantilla correspondiente que ha sido creada previamente por el administrador. Esta plantilla de trabajo contiene un título, descripción por sección, campo de escritura

como se aprecia en la Figura 3 y al hacer clic sobre una de sus secciones en donde dice “clic aquí para agregar texto al párrafo” se presenta el Editor Lingweb junto con el asistente de escritura como se puede ver en la Figura 6. Finalmente, las opciones de corrección ortográfica y sintáctica se encuentran disponibles en el editor avanzado Lingweb. Cada opción activa una interfaz que permite la revisión del texto según el botón seleccionado.

3. Resultados y discusión

Se realizaron pruebas funcionales de tipo unitario y de integración. Estas pruebas fueron ejecutadas por el equipo de desarrollo bajo un entorno local, no intervinieron usuarios finales. Primero se ejecutaron las pruebas unitarias evaluando las funcionalidades planteadas y posteriormente se verificaron que las funcionalidades examinadas en las pruebas unitarias funcionaran en su conjunto con todo el proyecto Lingweb.

Además, se utilizaron listas de chequeo como instrumento para la verificación de los requisitos planteados para el desarrollo del asistente (Rivera,

2011). Estas se crearon a partir de los requisitos funcionales a evaluar y estuvieron formadas por una serie de ítems expresados en forma de preguntas que permitieron confirmar la aceptación o no del cumplimiento del requisito. Para el caso del asistente, se diseñaron cinco listas de chequeo, un ejemplo de ellas se aprecia en la Figura 7. De los valores obtenidos en estas listas de chequeo se pudo evidenciar que el prototipo del asistente cumple con las funcionalidades establecidas para los procesos de acompañamiento y revisión de errores en los ejercicios de escritura de párrafo.

Las pruebas con usuarios finales están en progreso, así como el diseño de la experimentación que estructurará y orientará su desarrollo. Se espera evaluar con estas pruebas el grado de percepción de los estudiantes y profesores hacia el asistente y cómo éste puede contribuir en los procesos de enseñanza aprendizaje en la producción de textos.

4. Conclusiones y trabajo futuro

Con la arquitectura propuesta se establece un modelo genérico para la asistencia virtual en producción textual, ya que tiene en cuenta los



Figura 6. Interfaz asistente textual

LISTA DE VERIFICACIÓN		
DETALLE	RESULTADO	
1. El editor permite la revisión ortográfica.	SI <input checked="" type="checkbox"/> NO <input type="checkbox"/> PENDIENTE <input type="checkbox"/>	
2. El editor guarda los cambios realizados durante la corrección ortográfica.	SI <input checked="" type="checkbox"/> NO <input type="checkbox"/> PENDIENTE <input type="checkbox"/>	
3. El editor permite análisis sintáctico de oraciones.	SI <input checked="" type="checkbox"/> NO <input type="checkbox"/> PENDIENTE <input type="checkbox"/>	
4. El editor guarda las correcciones sintácticas realizadas.	SI <input checked="" type="checkbox"/> NO <input type="checkbox"/> PENDIENTE <input type="checkbox"/>	
5. El editor permite el marcado general de errores.	SI <input checked="" type="checkbox"/> NO <input type="checkbox"/> PENDIENTE <input type="checkbox"/>	
6. El editor permite el marcado de errores por categorías.	SI <input checked="" type="checkbox"/> NO <input type="checkbox"/> PENDIENTE <input type="checkbox"/>	
7. El editor permite marcar errores por la categoría Grammar.	SI <input checked="" type="checkbox"/> NO <input type="checkbox"/> PENDIENTE <input type="checkbox"/>	
8. El editor permite marcar los errores por la categoría Vocabulary.	SI <input checked="" type="checkbox"/> NO <input type="checkbox"/> PENDIENTE <input type="checkbox"/>	
9. El editor permite marcar los errores por la categoría Syntax.	SI <input checked="" type="checkbox"/> NO <input type="checkbox"/> PENDIENTE <input type="checkbox"/>	
10. El editor permite comentar una sección del texto.	SI <input checked="" type="checkbox"/> NO <input type="checkbox"/> PENDIENTE <input type="checkbox"/>	
11. El editor permite comentar de forma general un texto.	SI <input checked="" type="checkbox"/> NO <input type="checkbox"/> PENDIENTE <input type="checkbox"/>	
COMENTARIOS		

Figura 7. Ejemplo lista de chequeo

elementos léxicos, reglas ortográficas, reglas gramaticales, y recomendaciones de escritura que dependiendo del idioma se pueden ir actualizando o adicionando, esto permite una extensión del modelo según las necesidades de cada lengua.

El asistente virtual brinda su acompañamiento en la producción textual por medio de sus recomendaciones y afianza a Lingweb como un ambiente virtual de aprendizaje rico en nuevas estrategias metodológicas para la enseñanza de un segundo idioma.

La fusión de las características de agentes de software, ambientes virtuales de aprendizaje y elementos de la lingüística computacional robustece al asistente y facilita el procesamiento de lenguaje y las recomendaciones ofrecidas a los estudiantes en su proceso de escritura.

Se proyecta mejorar los alcances del agente desarrollado, llevándolo a un nivel de análisis semántico, en donde este pueda conocer el significado de lo que el usuario está escribiendo y con base en esto realizar las posibles recomendaciones. Además, el prototipo puede crecer en el uso de otros idiomas puesto que los analizadores y el lexicón fueron creados de tal forma que permite la inserción de nuevos lexemas y unidades sintácticas.

También se puede considerar la posibilidad de realizar el acompañamiento al usuario en otro tipo de actividades, como el ejercicio tipo texto que trabaja con composiciones textuales de mayor nivel por

ejemplo: ensayos, discursos, textos informativos, entre otros.

5. Referencias bibliográficas

Aricapa, J., & Puentes, J. (2009). *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, Cali, Colombia.

Berdugo, M., & Pedraza, N. (2005). *Construcción de un Ambiente Web para la enseñanza y el aprendizaje de idiomas*. Cali: Universidad del Valle.

Berdugo, M., Herrera, O., & Valdiri, V. (2010). El desarrollo de la escritura académica en el ambiente virtual Lingweb: realidades y desafíos. *Lenguaje* 38(2d), 351-386.

Berdugo, M., & Pedraza, N. (2008). Evaluación de Lingweb: ambiente virtual para el aprendizaje de lenguas. *Lenguaje*, No 36 (2d), 473-509

Blackboard. Course Management System (CMS) [en línea]. [Consultado el 26 de septiembre de 2012] Disponible en: <<http://www.blackboard.com>>

Damerau, Fred J. (1964). A technique for computer detection and correction of spelling errors. *Communications of the ACM* 7(3), 171-176.

- Dotlrn. Learn, Research, Network. [en línea]. [Consultado el 17 de octubre de 2012] Disponible en <<http://dotlrn.org>>
- Gómez, J. (2002). *Modelado de Sistemas Multi-Agente*. Tesis doctorado. Departamento de Sistemas Informáticos y Programación, Facultad de Informática, Universidad Complutense de Madrid, España
- Holmes, M. *Markin*. [Consultado el 17 mayo de 2011] Disponible en: < <http://www.cict.co.uk/markin/index.php> >
- JADE (Java Agent Development Framework). [Consultado el 26 de septiembre de 2012] Disponible en:<<http://jade.tilab.com/>>
- JavaScript SpellCheck, Sitio Oficial. [Consultado el 17 de junio de 2011] Disponible en: <http://www.javascriptspellcheck.com>
- Laszlo System. Sitio oficial de Openlaszlo. [Consultado el 26 de septiembre de 2012] Disponible en:<www.openlaszlo.org>
- Machuca, L. (2010). *Arquitectura multiagente para sistemas e-learning centrados en la enseñanza de idiomas (SE-MAS)*. Tesis de Maestría, Escuela de Ingeniería de Sistemas y Computación, Universidad del Valle, Cali, Colombia.
- Machuca, L., & Rodríguez, P. (2011). Experiences in developing an e-learning system for language learning and teaching. *IEEE Xplore Digital Library* [en línea]. [Consultado el 25 de junio de 2012] Disponible en: <http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=5936281>
- McRoy, S. W.; Channarukul, S. & Ali, S. S. (2001), 'Creating Natural Language Output for Real-Time Applications', *ACM Intelligence* 12 (2) , 21-34 .
- Marzal, A., & Vidal, E. (1993). Computation of Normalized Edit Distance and Applications. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 15(9), 926-932.
- Mas, A. (2005). *Agentes Software y Sistemas Multi-Agente: Conceptos, Arquitecturas y Aplicaciones*.
- Madrid: Pearson Prentice Hall. ISBN 84-205-4367-5.
- Moodle. Course Management System (CMS)[en línea]. [Consultado el 26 de septiembre de 2012] Disponible en:< <http://moodle.org>>
- Pavón, J, & Gómez, J. (2003). Agent oriented software engineering with INGENIAS. In: V. Marik, J. Muller, & M. Pechoucek, *Multi-Agent Systems and Applications III*, Springer Verlag (volume 2691 of LNCS., 394–403).
- Peña, C. (2000). *Sistemas Multiagente para el Tratamiento de la Información en el WEB: Agentes de Interfaz, Agentes de Información, Agentes de Aprendizaje, Agentes Intermediarios*. En I Congreso Internacional de Ingeniería de Sistemas - EISI 30 años, Universidad Industrial De Santander, Bucaramanga, Colombia. [Consultado el 14 de febrero de 2012]. Disponible en <<http://eia.udg.es/~clarenes/pdfs/publicaciones/congresosuis-2000-agentes-cip.pdf>>
- Rivera, D. (2011). *Asistente virtual para la producción textual (AVPT)*. Tesis de Maestría, Escuela de Ingeniería de Sistemas y Computación, Universidad del Valle, Cali.
- Ross, S., Brownholtz, E., & Armes, Robert. (2004). *A multiple-application conversational agent*. In Proceedings of the 9th international conference on Intelligent user interfaces IUI '04. Publisher: ACM Press. Funchal, Portugal
- Sakai Collaboration and Learning Environment (CLE) [en línea] [Consultado el 8 de octubre de 2012] Disponible en: <<https://sakaiproject.org>>
- Sidner, C., Boetter, C., & Rich, C. (2000). *Lesson Learned in Building Spoken Language Collaborative Interface agents*. In Proceedings of the ANLP-NAACL 2000 Workshop on Conversational Systems, p. 6.
- Traum, D., & Rickel, J. (2002). *Conversational agents: Embodied agents for multiparty dialogue in immersive virtual worlds*. In Proceedings of the first international joint conference on Autonomous agents and multiagent systems (AAMAS): part 2. Publisher: ACM Press. Bolonia, Italia.