

UNIVERSIDAD AUTÓNOMA DE BAJA CALIFORNIA
FACULTAD DE INGENIERÍA, ARQUITECTURA Y DISEÑO

ÁREA DE POSGRADO

MAESTRÍA Y DOCTORADO EN CIENCIAS E INGENIERÍA



**“AGENTE DE MEMORIA PARA EL APOYO EN SESIONES DE ESTUDIO
INDIVIDUAL”**

Realizado por:

GUILLERMO ALEJANDRO CHAVEZ SANCHEZ

Ensenada, B. C.

Enero del 2014

UNIVERSIDAD AUTÓNOMA DE BAJA CALIFORNIA

**FACULTAD DE INGENIERÍA, ARQUITECTURA Y DISEÑO
UNIDAD ENSENADA**

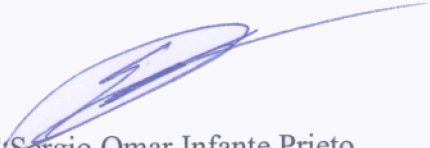
**AGENTE DE MEMORIA PARA EL APOYO EN SESIONES DE ESTUDIO
INDIVIDUAL**


TESIS


Que para obtener el grado de maestría en ingeniería presenta:

Guillermo Alejandro Chávez Sánchez

Aprobada por:


M.C. Sergio Omar Infante Prieto
Director de tesis


Eduardo Ceseña Beltrán
Miembro del comité


Manuel Jiménez Orozco
Miembro del comité

Ensenada Baja California, México. Enero 2014

AGRADECIMIENTOS

Le agradezco a mi madre (Noemí Sánchez Arriaga) por apoyarme en todas mis etapas de superación, soportar mis desveladas mis ausencias y comprenderme siempre. A mi hija (Magali Noemí) por comprender mi ausencia, y siempre estar al pendiente de mí.

Al maestro Christian X. Navarro Cota por creer en mí y apoyarme al inicio de mi investigación.

A mi Director de tesis Sergio Omar Infante Prieto por su apoyo y confianza al realizar esta tesis.

A mi sinodales Eduardo Ceseña y Manuel Jiménez por apoyarme en todo momento y cada etapa de mi investigación.

Gracias a mis compañeros del CUBO por su amistad, consejos, buenos momentos y muchas historias que vivimos juntos en todo este tiempo (Javier, Francisco, José María, Daniel, Diego, Ayala, Alan, Raymundo, Parma, Ismael, Catalina).

A mis ex compañeros de trabajo que siempre me apoyaron en todas las etapas de esta tesis (Oscar, Guadalupe, Rosy, Rosi, Salvador, Ruben, Temmy, Jacky, Vicky, Betty, Hugo, Laura, Denis, Linda.

A Conacyt por otorgarme la beca para poder realizar mi superación profesional con esta maestría.

Contenido

1	Introducción	1
1.1	Planteamiento del Problema	5
1.2	Planteamiento de la Solución	7
1.3	Objetivos	11
1.4	Limitaciones	12
1.5	Terminología	12
1.6	Estado del Arte	14
2	Agente de Memoria	21
2.1	Arquitectura	22
2.2	Implementación	24
2.3	Requerimientos	26
2.4	Módulos	26
2.4.1	AgenteLibreria.pm	29
2.4.2	Purifica_Cadena	29
2.4.3	Separa_Cadena	29
2.4.4	Calcula_Tema	31
2.4.5	Busca_Subtema	31
2.4.6	Top_Subtema	31
2.4.7	StopList.pm	32
2.4.8	Recomendador.pm	33
2.4.9	ConectarDB.pm	37
2.4.10	Agente.cgi	37
2.4.11	Agente.pl	40
2.4.12	Carga-Tema.pl	40
2.4.13	Base de datos	41
2.4.14	Protocolo de Comunicación con el Agente	44
2.4.15	Modelo de Comunicación con el Agente	45
2.4.16	Formato	47
2.5	Editor de Texto (WEB)	48
2.6	Editores de contenido	49
2.6.1	Editor StopList	49
2.6.2	Editor de Palabras en Subtemas	51
2.6.3	Indexar un Archivo	53

2.7 Requerimientos para instalación.....	54
3 Pruebas.....	56
3.1 Resultados de las pruebas	61
4 Conclusiones, Aportaciones y Trabajo a Futuro	64
5 APÉNDICE A: Guía Instalación del Sistema.....	65
6 Referencias.....	66

Lista de Figuras

Figura 1: Diagrama de los elementos del sistema propuestos.....	8
Figura 2: Emacs y su sistema de realidad aumentada.....	15
Figura 3: Realidad Aumentada en la Enseñanza de las Matemáticas.....	16
Figura 4: Magic Book.....	17
Figura 5: Realidad Aumentada en diversas áreas.....	18
Figura 6: Arquitectura propuesta.....	24
Figura 7: Diagrama de relación de los paquetes del Agente de Memoria.....	28
Figura 8: Diagrama de flujo del Agente de Memoria.....	30
Figura 9: Funcionamiento del StopList.....	33
Figura 10: Diagrama de Flujo para la recomendación de una palabra.....	34
Figura 11: Diagrama de vínculos entre Temas y Subtemas.....	36
Figura 12: Diagrama de flujo del Agente usando todos sus módulos.....	38
Figura 13: Panel de administración de las tablas del sistema.....	41
Figura 14: Tablas y su contenido.....	43
Figura 15: Diagrama que muestra la forma de comunicarse con el Agente.....	46
Figura 16: Editor de texto para el Agente.....	49
Figura 17: Modulo agregar palabras a StopList.....	50
Figura 18: Editor de contenido de StopList.....	51
Figura 19: Editor Subtemas.....	52
Figura 20: Modulo indexar un archivo.....	53
Figura 21: Tabla de análisis para "La media".....	58
Figura 22: Tabla de Estadística.....	62

1. Introducción

Una parte importante del proceso de aprendizaje requiere que el estudiante estudie por cuenta propia, que repase material adicional al visto en la clase y que investigue un poco más, no todo lo ve en el salón de clases. El alumno promedio al leer un libro, tarea o publicación no cuenta con la buena costumbre de utilizar un diccionario para consultar significados de las palabras no comprendidas. También frecuentemente hace caso omiso a la información presentada en forma más gráfica como diagramas, tablas, etc. Como consecuencia, algunos estudiantes después de haber leído el texto por más de un par de ocasiones caen en cuenta que no han leído con la atención necesaria para la comprensión del texto. Existen materias las cuales requieren de textos cuya lectura es difícil pero nunca está de más que los docentes se aseguren de recomendar textos de fácil comprensión. Además deben animar a sus alumnos a utilizar desde muy temprano en su carrera estrategias que les ayuden a dominar el vocabulario y las formas de expresión propia de la profesión [1].

Lockwood, Williams y Roberts (1988) [23] señalan que el tiempo que un estudiante requiere para estudiar está determinado por su velocidad de lectura en relación con su comprensión. Por su parte Chambers (1992) [24] dice que esto tiene que ver con el grado de dificultad que representa cada texto. Los

textos más fáciles se pueden leer a 100 palabras por minuto, mientras que los más complicados pueden reducir la velocidad a unas 40 palabras por minuto. Estudios realizados por Blacklock en la Universidad de Londres (1976) y McKay (1978) [25] en la Universidad de Canterbury sugieren que un estudiante de tiempo completo a nivel de licenciatura dedica alrededor de 40 horas semanales a su programa académico y que el estudiante promedio dedica alrededor del 50% de este tiempo en estudio privado cumpliendo con los requerimientos de clase, reforzando y clarificando las ideas así como preparándose para evaluaciones. También señala que el tiempo dedicado a las clases formales varía entre 14 y 20 horas semanales según el tipo de carrera. Según Gordon (1996) un estudio realizado por Michael Lacopo entre universidades en Estados Unidos señala que el tiempo dedicado a estudiar fuera del aula se ubica en un rango entre 30 minutos hasta un máximo de 75 minutos por cada clase que se recibe [1].

Respecto a las conductas relacionadas con tomar notas, lo cual determina en gran medida la calidad del aprendizaje del alumno, se observa que, aunque no se trata de conductas practicadas de manera generalizada es la que mejores resultados positivos obtiene. Otras conductas utilizadas son el uso de palabras clave y resúmenes para abreviar las notas y la utilización de palabras propias

para dar a entender las palabras exactas del maestro [1].

Los procesos de enseñanza y aprendizaje poco a poco se han visto enfrentados a transformaciones significativas ocasionadas por los cambios tecnológicos, que imponen, tanto a profesores como a alumnos, el desarrollo de nuevas competencias que les permiten nuevas experiencias para la comprensión de los conceptos objeto de estudio [2].

Por otro lado, el avance tecnológico de los dispositivos móviles (Celulares, Tablet, etc.) ha permitido crear nuevas formas de ver nuestra vida cotidiana. Algunos años atrás estos dispositivos solo permitían realizar llamadas, tomar fotografías, el envío de mensajes de texto y realizar otras tareas simples. Actualmente, los dispositivos móviles son capaces de realizar las tareas que antes se creía solo podían ser realizadas en computadoras personales o de escritorio. Gracias a estos avances han emergido infinidad de sistemas y proyectos de gran alcance. Actualmente existen dispositivos móviles capaces de procesar imágenes, realizar tareas multimedia, ubicar nuestra posición y orientación por medio de GPS, Acelerómetros y Brújulas.

Hasta hace poco, las computadoras solo habían tenido acceso al contexto

actual de un usuario dentro de una tarea determinada. Por ejemplo el procesador de texto tiene acceso a lo escrito en el documento actual pero no puede acceder a otros archivos para aportar información al nuevo documento. De igual manera, tampoco puede saber qué tipo de trabajo está realizando el usuario, su posición actual, si esta solo o en grupo, hablando o leyendo, etc.

Actualmente para dar un mejor apoyo al aprendizaje es necesario reunir y recordar notas, libros, correos, fechas, nombre de archivos, pláticas, etc. Esto se hace de una forma física, no muy ordenada y tener acceso rápido a esta información es difícil. Existen sistemas como las agendas electrónicas, buscadores de archivos por semántica y sistemas inteligentes para predecir lo que el usuario necesitaría para realizar su trabajo. Una propuesta son los Agentes de Memoria en conjunto con la Realidad Aumentada. Un Agente de Memoria es un programa que continuamente "mira por encima del hombro" del usuario de una computadora portátil y muestra los resúmenes de una línea de notas, archivos, correo electrónico, documentos y demás información de texto que pueda ser relevante al contexto actual del usuario [4]. Como apoyo al aprendizaje, un Agente de Memoria podría estar mirando por encima del hombro de un estudiante, mientras este se encuentra en su sesión de estudio, y presentar información que pudiera ser útil para la comprensión del tema que el estudiante está repasando. Por otro lado, la Realidad Aumentada es una

tecnología que complementa la percepción e interacción con el mundo real y permite al usuario estar en un entorno real aumentándolo con información adicional generada por la computadora [3]. En un sistema de apoyo al aprendizaje, por medio de Realidad Aumentada podría presentarse la información adicional encontrada por el Agente de Memoria, integrándola al entorno actual del estudiante.

1.1 Planteamiento del Problema

No existe un sistema que apoye a las sesiones de estudio individual, que ayude a mantener las costumbres de estudio planteadas en la sección anterior “Estudiar por cuenta propia, Repaso de material, Investigación extra, Utilización de diccionarios, etc.”. Sería conveniente contar con un sistema basado en Realidad Aumentada que se ubique en el contexto de estudio del usuario.

Actualmente existen sistemas de Realidad Aumentada orientados al aprendizaje, por ejemplo “Magic Book” [3], que muestra objetos virtuales para enriquecer el contenido, sin embargo los objetos virtuales mostrados no cambian con el aprendizaje o avance del usuario. Esto lleva a que siempre se muestre la misma información estática sin actualizarse o sugerir nuevo

material. Otros sistemas hacen búsquedas en Internet/Bases de Datos y de igual manera, la información que el sistema presenta sigue sin aportar nada que el usuario no pudiese encontrar por el mismo. Para aumentar el contenido, estos sistemas se basan particularmente en marcadores, los cuales aportan posición en el mundo real en la cual se debe desplegar la información que aumenta el contenido, así como una referencia con la cual se obtiene la información que pertenece a dicho marcador.

La falta de un sistema para el apoyo del aprendizaje se ve reflejada al momento de la consulta o realización de tareas. Normalmente una persona lee más de una sola vez un documento para revisar su contenido. También existe pérdida de notas y pérdida de tiempo para recordar el nombre de los documentos o fechas de los correos sobre un tema específico que podría ser crucial para el aprendizaje que se lleva a cabo.

1.2 Planteamiento de la Solución

Con la tecnología existente en la actualidad se podría crear una solución basada en Realidad Aumentada y dispositivos móviles, la cual servirá de apoyo al aprendizaje en sesiones de estudio del usuario. Este sistema captaría la información del texto que el usuario estuviese leyendo y la aumentaría en forma de vínculos sobre el tema presentando. Dichos vínculos presentarían Páginas Web, Vídeos, Audio, Archivos de texto, etc., y serán encontrados automáticamente en base al contenido del texto y los objetivos de aprendizaje del tema que se está consultando. Los principales elementos del sistema de Realidad Aumentada planteado se presentan en el siguiente (Figura 1).

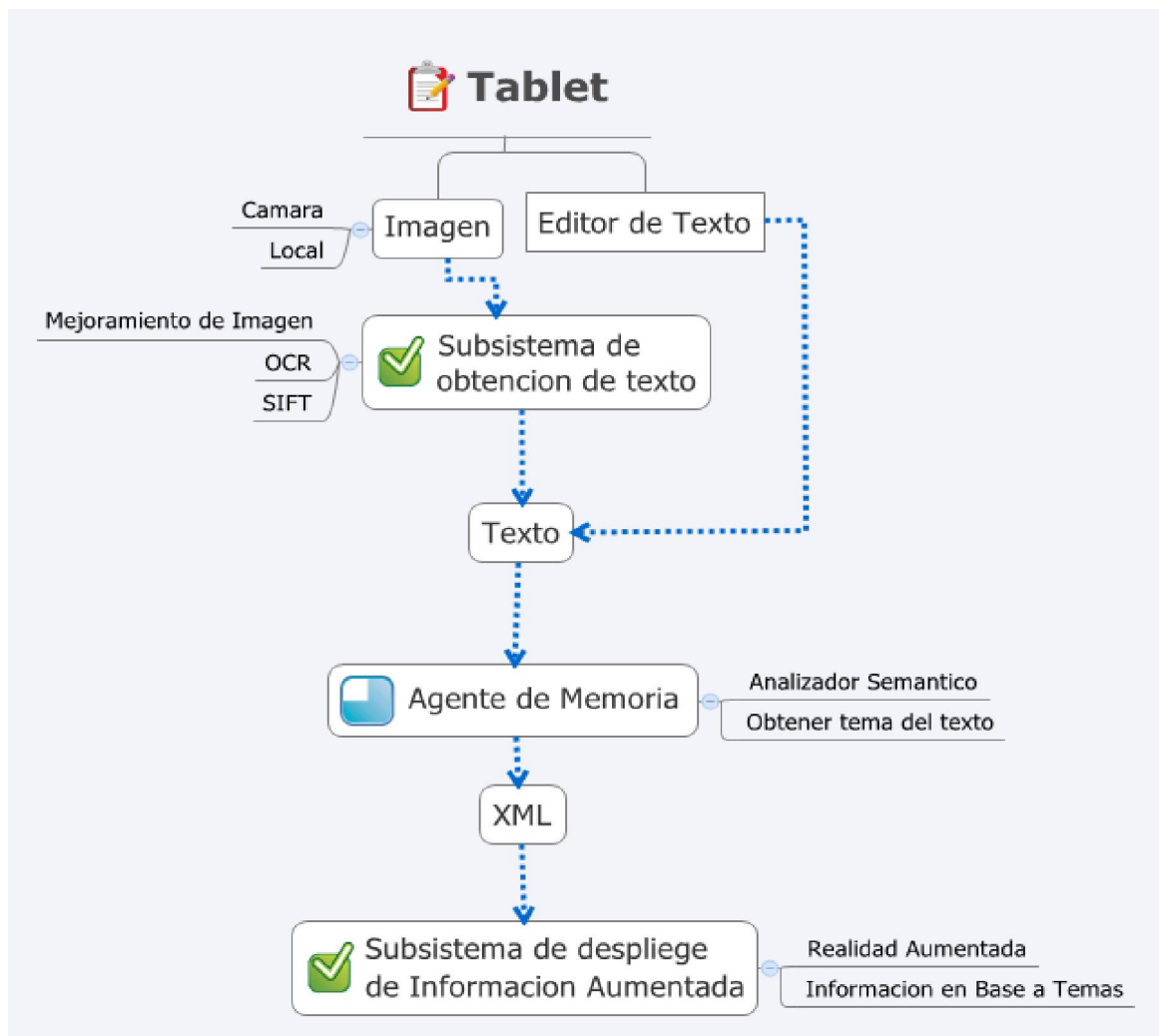


Figura 1: Diagrama de los elementos del sistema propuestos.

Cada elemento de este sistema se describirá a continuación:

- **Tablet** o cualquier otro dispositivo móvil que el estudiante pueda utilizar para sus sesiones de estudio individual. Este dispositivo debe tener acceso a internet y contar con una cámara. Su tarea es obtener la información a analizar. Dicha información puede provenir de un procesador de textos o puede ser una imagen del material de estudio adquirida por medio de la cámara del

dispositivo.

- **Mejoramiento de Imagen.** Este módulo procesará la imagen adquirida para mejorarla y adaptarla para la extracción de la información contenida en la imagen.
- **OCR/SIFT** es un módulo que aplicará un conjunto de tecnologías que se encargan de extraer el texto y figuras de las imágenes. Se obtiene el texto en forma de palabras, números y signos, los cuales pasan a un análisis para solo usar información de valor.
- **Analizador Semántico** este se encarga de analizar y comparar las palabras obtenidas del procesador de textos o de la imagen. Su tarea es encontrar temas relacionados con esas palabras para poder elegir y valorar los temas encontrados. Una vez encontrados los temas se crea una lista con sus respectivos valores y son enviados al agente de memoria.
- **Agente de Memoria** busca y clasifica las sugerencias/recomendaciones para la lista de temas. Estas son ordenadas por importancia o relevancia en el texto y otros aspectos en su análisis. La información puede provenir de otros archivos, imágenes, videos, figuras 3D y más.
- **Realidad Aumentada** es modulo que se encarga de generar la información obtenida con el Agente de Memoria mezclándola con la información original. Esta información puede ser en forma de texto anexo, de

videos, estructuras 3D animadas o estáticas, todo esto para enriquecer la información que el usuario está trabajando.

- **Información Aumentada** es la mostrada en pantalla, la cual es la combinación de la información original con la recomendada por el agente, y que ha sido generada por el módulo de realidad aumentada.

En este proyecto nos enfocaremos a resolver el sistema de Agente de Memoria. Actualmente existe un agente de memoria que solo corre en el procesador de textos Emacs, está orientado a personas con conocimientos previos de este procesador de textos y no al usuario común, cuenta con una interfaz no intuitiva para los usuarios menos expertos, solo trabaja con información en archivos locales del dispositivo en el que se encuentra ejecutándose y corre en computadoras laptop o de escritorio [6].

Dentro de la realización de este sistema se encuentran los siguientes retos:

- a) Implementar el concepto actual de Agente de Memoria para diseñar uno de acuerdo a las necesidades en el aprendizaje individual.
- b) Creación de una interfaz intuitiva para comunicar al Agente de Memoria con

el usuario.

c) Superar las limitaciones de prestaciones en los equipos Tablet: Procesamiento, Memoria, Capacidad de Gráficos, Calidad de cámara.

d) La correcta fusión y elección de las tecnologías involucradas para obtener el resultado esperado.

Se tiene planeado un sistema escalable, esto facilitaría una forma flexible de mejorar o aportar al proyecto.

1.3 Objetivos

El objetivo general del proyecto es apoyar al aprendizaje por medio de un sistema de Realidad Aumentada en Tablet que sirva de apoyo en sesiones de estudio individual.

También se tienen los siguientes objetivos específicos:

- Crear una interfaz para el sistema intuitiva y amigable para el usuario con respecto al uso de los Agentes de Memoria, para mejorar la comunicación entre el usuario y el sistema.
- Diseñar un Agente de Memoria para el aprendizaje acorde a las nuevas

tecnologías.

- Dar a conocer las ventajas y potencialidad de los Agentes de Memoria.

1.4 Limitaciones

En este trabajo solo se nos enfocamos en generar el Agente de Memoria, debido a lo extenso del tema y desarrollo, la implementación del sistema se realizó con un editor de texto en la web para probar la eficiencia del Agente, otras limitaciones el desarrollo de la investigación fueron:

- Falta de una base de datos con las palabras importantes para cada subtema.
- Tiempo para contactar expertos en los diferentes Temas y subtemas para poder delimitar las palabras importantes y su valor.
- Tiempo para desarrollar una versión para dispositivos móviles, para realizar pruebas más reales.

Las pruebas realizadas para el sistema serán de un corto alcance, considerando solamente un par de temas de la carrera de Ingeniería. Se analizará texto de tareas, exámenes, apuntes y libros de uso común en la carrera.

1.5 Terminología

A continuación se presenta algunos términos o definiciones mencionados en

este trabajo:

Perl: Lenguaje Práctico para la Extracción e Informe [13].

REST: Transferencia de Estado Representacional [14].

MySQL: Sistema de gestión de bases de datos relacional multi-hilo y multiusuario [15].

RA: Realidad Aumentada [3].

QR: Abreviatura de “código de respuesta rápida” [25].

Open Source: Es la expresión con la que se conoce al software distribuido y desarrollado libremente [26].

GPL: Son las siglas de “Licencia Pública General de GNU” [27].

PHP: Lenguaje de programación de uso general de código del lado del servidor [16].

CGI: Interfaz de entrada común (en inglés *Common Gateway Interface*) es una importante tecnología de la World Wide Web que permite a un cliente (navegador web) solicitar datos de un programa ejecutado en un servidor web [20].

SIFT: Transformada de Rasgos de Escala Invariable (o SIFT) es un algoritmo de visión artificial para detectar y describir las características locales de las

imágenes [21].

OCR: Reconocimiento óptico de caracteres [22].

XML: Lenguaje de marcas extensible [17].

StopList: En computación una StopList es el conjunto de Palabras Vacías, estas son palabras sin significado alguno para el propósito del sistema por ello son filtradas antes de procesar los datos en lenguaje natural. El concepto de Palabras Vacías se le acuñó a Hans Peter Luhn [11].

Servidor HTTP: Es un programa informático que procesa una aplicación del lado del servidor realizando conexiones bidireccionales y/o unidireccionales y síncronas o asíncronas con el cliente generando o cediendo una respuesta en cualquier lenguaje [28].

1.6 Estado del Arte

El proyecto de los laboratorios del MIT “Augmented Reality Through Wearable Computing” [6], trata de adaptar al usuario una computadora, para poder analizar su entorno y enriquecerlo por medio del análisis de la esta información en tiempo real, esta es mostrada con realidad aumentada ver [Figura 2].

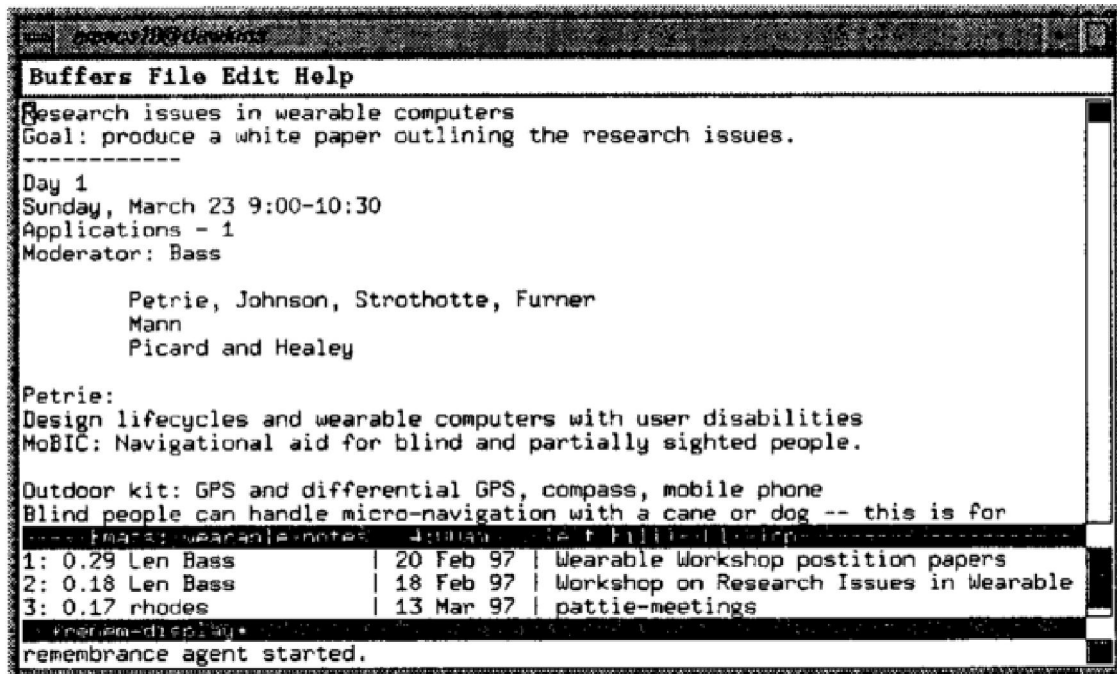


Figura 2: Emacs y su sistema de realidad aumentada.

Un objetivo a largo plazo de este proyecto es modelar las acciones del usuario, anticiparse a sus necesidades y apoyarlo en la interacción entre los entornos físicos y virtuales.

En el Laboratorio de Realidad Virtual de la Universidad Eafit, se desarrolló un prototipo de Realidad Aumentada para la enseñanza del Cálculo en Varias Variables, que permite la visualización de conceptos matemáticos a partir de la creación de un objeto virtual, que se puede comparar con objetos reales, potenciando las posibilidades de comprensión de los conceptos matemáticos estudiados [2].

El Institut Graphische Datenverarbeitung (IGD) investiga aplicaciones de Realidad Aumentada para entrenamiento industrial, diseño de interiores, supervisión de plantas, visualización de datos, construcción exterior y técnicas de ensamble de máquinas [2].

El proyecto de investigación “Realidad Aumentada en la Enseñanza de la Matemática”, teniendo como base pedagógica la Enseñanza para Comprensión, se diseñó un software especializado, que le permite al profesor y a un alumno interactuar y visualizar [7] superficies en 3D generadas por la computadora, a través de una cámara de vídeo y de unas gafas de realidad aumentada sobre una superficie real, permitiendo la comparación del objeto virtual con objetos de la realidad.

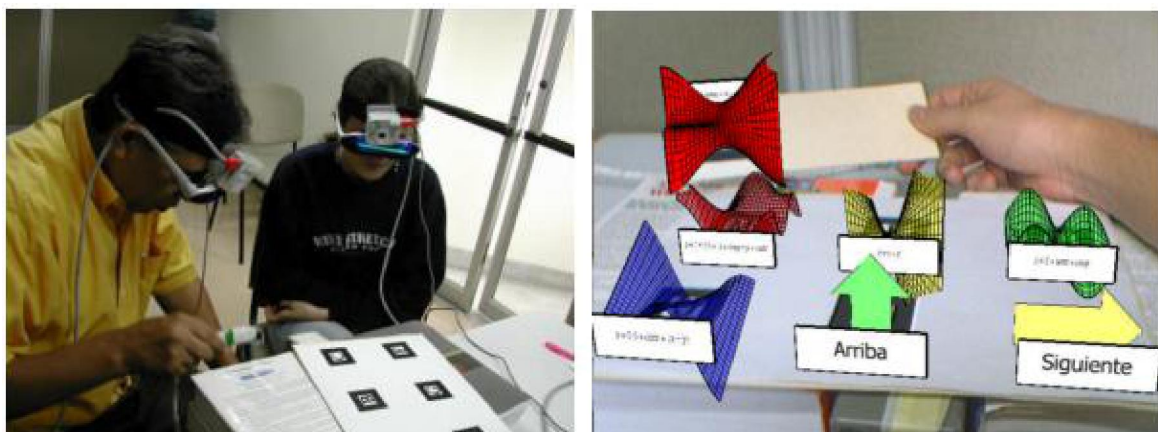


Figura 3: Realidad Aumentada en la Enseñanza de las Matemáticas.

Quizá una de las aplicaciones más conocidas de la Realidad Aumentada en la educación sea el proyecto Magic Book [3] del grupo activo HIT de Nueva Zelanda. El alumno lee un libro real a través de un visualizador de mano y ve sobre las páginas reales contenidos virtuales. De esta manera cuando el alumno ve una escena de Realidad Aumentada que le gusta puede introducirse dentro de la escena y experimentarla en un entorno virtual inmerso [3].

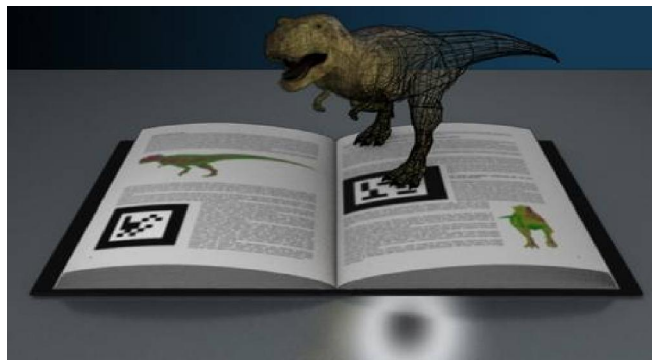


Figura 4: Magic Book.

Instituciones de prestigio como el Massachusetts Institute of Technology (MIT) y la Universidad de Harvard están desarrollando en sus programas y grupos de Educación aplicaciones de Realidad Aumentada en formato de juegos; estos juegos buscan involucrar a los estudiantes de educación secundaria en situaciones que combinan experiencias del mundo real con información adicional que se les presenta en sus dispositivos móviles [8].

También han desarrollados juegos para enseñar materias de matemáticas y

ciencias, y todos ellos están orientadas a trabajar de forma colaborativa entre los estudiantes.

En el ámbito europeo existen diferentes proyectos que diseñan y desarrollan aplicaciones innovadoras que integran Realidad Aumentada para ser utilizadas en la educación. Entre otros proyectos se pueden destacar CONNECT, CREATE y ARiSE [3]. Estas nuevas herramientas basadas en presentaciones 3D y con gran interacción facilitan la comprensión de las materias de todas las ciencias. Los estudiantes pueden interactuar con objetos virtuales en un entorno real aumentado y desarrollan el aprendizaje experimentando.



Figura 5: Realidad Aumentada en diversas áreas.

Todos estos proyectos e iniciativas están basadas en Realidad Aumentada por medio de marcadores de RA o bien códigos QR, esto limita mucho el aprovechar el materia ya existente en nuestras bibliotecas y/o hogares, material el cual puede ser aprovechado con el apoyo de las tecnologías OCR y Realidad

Aumentada.

Los sistemas anteriores en su mayoría utilizan equipo muy costoso y algo incómodo para el uso común de las personas, actualmente con el poder de computo de los dispositivos móviles Celulares, PDA, Laptop, Tablet...etc., los cuales cuentan con videocámara, reproducción multimedia y una conexión a internet y más comunes entre la comunidad estudiantil, se puede hacer llegar esta tecnología a sus manos de una manera fácil y barata.

Dentro de los sistemas que no utilizan marcadores RA, códigos QR se encuentra goggles de la empresa Google. Esta aplicación permite hacer búsquedas en Google a partir de una imagen tomada con la cámara del teléfono, cuenta con reconocedor de texto "OCR" al igual te permite hacer la búsqueda o traducir el texto o usarlo en alguno de los servicios de Google [9].

Alchemy Snap es una aplicación de Android que tampoco utiliza marcadores. Esta aplicación identifica el texto en una hoja de papel o pantalla de una computadora por medio de OCR. El texto obtenido puede ser usado para hacer búsquedas en varios servicios "Google, Twitter, Wikipedia, etc...", por lo cual se dice que cuenta con búsqueda semántica, al igual que muchos no soporta que el papel u objeto sea en forma curva o tenga sombreados por falta de luz,

no identifica OCR en superficies que no sean de un fondo blanco o muy claro a comparación a la letra. No identifica formulas o nada que no sea del alfabeto y números [10].

El potencial futuro de la Realidad Aumentada es muy grande. No tanto por lo que la tecnología puede hacer en sí, sino por su capacidad para mejorar la experiencia de otras tecnologías actuales y futuras. Ahora llevamos dispositivos GPS en el automóvil, pero en un futuro próximo el GPS vendrá de serie en todos los modelos, la información se mostrará en el parabrisas, e incluirá incidencias en la carretera, ofertas en tiendas cercanas adaptadas a nuestros perfiles, o avisos de gente conocida que se encuentre en los alrededores. Recorreremos los centros comerciales observando a través del móvil, o de unas gafas, los productos que vamos a comprar mientras se nos muestran opiniones de otros consumidores sobre ese producto, ofertas de productos similares o posibles usos para el mismo en los que quizás aún no hayamos pensado.

La realidad aumentada puede poner en nuestras manos, de una forma práctica, toda la información que tenemos a nuestra disposición en la era de Internet.

2 Agente de Memoria

Un Agente de Memoria se puede ver como una simulación de un asistente personal el cual está pendiente del entorno en el que estamos trabajando y la información que se está consultando en ese momento. Este asistente está pendiente de la información que generamos o consultamos, nos apoya con sugerencias de otras fuentes de información así como de recomendaciones para mejorar nuestro trabajo. Este asistente puede ser versátil y adaptarse a situaciones nuevas, puede recordar problemas pasados, recuperar información, analizar grandes cantidades de información y entregar resúmenes o referencias de esos análisis. Todo esto se puede lograr gracias a que el asistente, en base a palabras clave, puede realizar un mejor análisis para buscar en sus fuentes de información e entregar un mejor resultado.

En este capítulo se describe la arquitectura y requerimientos, así como la explicación de los módulos que componen al Agente de Memoria. También se explica el diseño para su desarrollo, de tal forma que cualquier persona pueda portarlo a la tecnología de su conveniencia. El sistema se comunica por su propio sistema de mensajes, por lo cual, no está apegado a ningún lenguaje o tecnología en concreto. Cualquier lenguaje que pueda realizar una conexión a un servicio WEB, podrá utilizar este Agente de Memoria.

La explicación y entendimiento de la relaciones entre recomendaciones/sugerencias del texto analizado es explicado en base a diagramas de relación, así se comprende de una manera más precisa el trabajo realizado con este proyecto.

2.1 Arquitectura

Después de analizar proyectos relacionados, se realizó esta propuesta de arquitectura para cumplir con las necesidades del agente de memoria y su interacción con el usuario final. Se decidió utilizar un sistema remoto el cual procese y devuelva las recomendaciones al usuario, esto con fin de tener un sistema en el que sea posible la escalabilidad. Esto quiere decir que se podrán agregar más fuentes de información y módulos de procesamiento para la mejora del mismo.

Para la arquitectura del sistema del Agente de Memoria, se proponen los siguientes componentes:

- **Servidor:** Sera el encargado de alojar al Agente de Memoria, Base de Datos, un Editor de pruebas y Herramientas Web para administrar la base de conocimiento del Agente. En este servidor también se alojarán los documentos, imágenes, videos y demás archivos utilizados.
- **Agente Memoria:** El agente recibe las peticiones a procesar por medio

del servidor. Cuenta con una serie de módulos para realizar las tareas de filtrado del texto, análisis sintáctico, acomodar y relacionar las palabras con sus temas correspondientes y regresa como resultado un archivo de respuesta. El Agente de Memoria podrá acceder a la base de datos y a los archivos guardados en el servidor.

- **Base de Datos:** En esta se almacenará la información para analizar el texto por parte del Agente, además de almacenarse nueva información sobre Temas y Subtemas analizados. La Base de Datos proporcionará acceso a la información de forma abierta para cualquier consulta externa.
- **Fuente:** Es la información que será enviada al Servidor que contiene al Agente. Esta información puede ser proporcionada por diversas fuentes, ya sea un editor de texto, un navegador Web, un teléfono móvil, etc.

Este esquema de trabajo se puede ver en la siguiente (Figura 6).

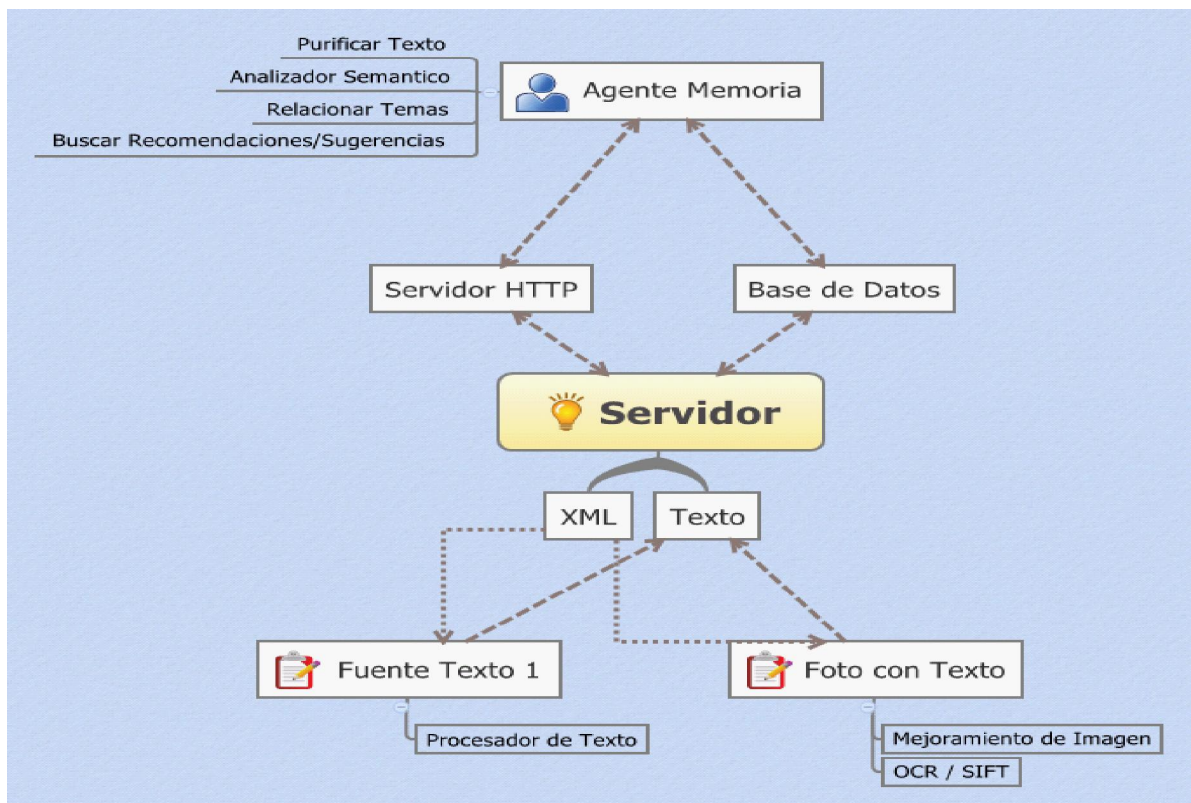


Figura 6: Arquitectura propuesta.

En caso necesario de requerir un mayor almacenamiento o una mejora al sistema, se pueden realizar cambios para crear un sistema en paralelo y así mejorar los tiempos de respuesta y análisis del sistema.

2.2 Implementación

En la Arquitectura se habló de que este Agente de Memoria sería un sistema escalable. Con este propósito se eligió una arquitectura Cliente-Servidor. De esta manera la carga del procesamiento que será realizado se divide en dos

partes. Una parte es el cliente que es el que solicita el análisis al Agente, esto lo hace empaquetando el texto a analizar y enviando directo al servidor donde se encuentra el Agente. En el otro extremo contamos con el servidor, este se encargará de recibir y analizar el texto enviado por el cliente, generar una respuesta en base a los criterios y análisis del Agente, la cual será devuelta al cliente en formato texto para su fácil procesamiento.

Para realizar las pruebas del sistema se realizó un editor web a la medida, el cual será explicado a detalle en la sección 2.6 de este capítulo (Editores de contenido). Los clientes se comunican con el servidor por medio de una arquitectura de paso de mensajes llamada “REST” [14]. Esta arquitectura permite el envío de texto por la internet de manera más fácil y permite recibir de igual manera la respuestas. La información devuelta por el servidor será en formato XML, este formato es soportado por la mayoría de los lenguajes actuales. En el servidor, el Agente está programado bajo el lenguaje Perl y la información será almacenada en una base de datos MySQL. Todo el sistema está basado en tecnologías Open Source.

2.3 Requerimientos

Los requerimientos del sistema por el lado del servidor son:

- Sistema Operativo para servidor.
- Soporte para el lenguaje de programación Perl.
- Soporte para el manejo y control de base de datos MySQL.
- Soporte protocolo HTTP y FTP.

Es necesario tener conocimientos básicos de servidores, para realizar pruebas cuando se instale el Agente. Este puede ser probado desde la línea de comandos del servidor. Para realizar pruebas el agente se instalo en un servidor con la dirección (<http://ia.ens.uabc.mx/>).

Los requerimientos del sistema por el lado del cliente son:

- Dispositivo con acceso a internet (Tablet, Teléfono, Laptop, Pc).
- Navegador Web (Opera, Firefox, Chrome,.. etc.).
- Opcionales Cámara y Bocinas.

Contando con estos requerimientos se puede trabajar con el Agente, para esto solo se necesita saber la dirección del servidor donde se encuentra alojado.

2.4 Módulos

En esta sección se describe la forma de operar del Agente de Memoria, así como los componentes del mismo. El diseño de código del Agente de Memoria

se divide en los siguientes paquetes, conteniendo cada uno sus módulos específicos:

AgenteLibreria.pm

- Purifica_Cadena
- Separa_Cadena
- Calcula_Tema
- Busca_Subtema
- Top_Subtema

StopList.pm

- stoplist_search

Recomendador.pm

- Sugerencia
- Recomendar_Web
- Recomendar_Video
- Recomendar_Imagen

ConectarDB.pm

- connect

Agente.cgi & Agente.pl

Este es nuestro CGI principal con el cual realizaremos las peticiones a los módulos del sistema para realizar las tareas de este Agente de Memoria.

La relación entre los paquetes y el CGI se puede observar en el siguiente (Figura 7).

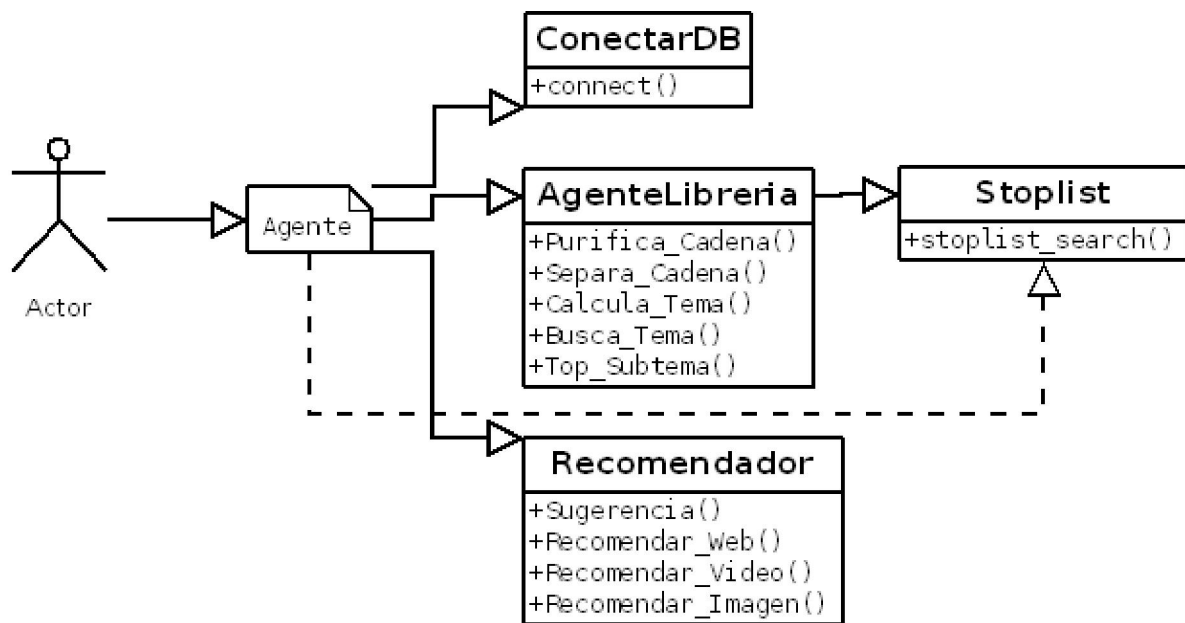


Figura 7: Diagrama de relación de los paquetes del Agente de Memoria.

El contar con un proyecto separado en paquetes nos da una manera más fácil de analizar y mejorar los módulos de forma individual o utilizarlos en otros proyectos.

A continuación se describe el funcionamiento de cada módulo y de los script principales para hacer funcionar el Agente de Memoria.

2.4.1 AgenteLibreria.pm

Este paquete contiene las funciones a realizar para el procesamiento del texto a analizar, así como todo lo relacionado con el cálculo y búsqueda de temas relacionados con las palabras. Los módulos que lo componen y su explicación es la siguiente:

2.4.2 Purifica_Cadena

Esta función recibe un texto lo separa en un arreglo de caracteres. Después filtra las palabras removiendo todo carácter no alfanumérico y convirtiendo todo a minúsculas. Al finalizar devuelve un arreglo ya purificado.

2.4.3 Separa_Cadena

Separa y contabiliza las palabras repetidas, verificando en una lista de palabras no válidas para el análisis del texto “StopList”, a su vez verifica si existen palabras compuestas en el texto. Este proceso se ilustra en la Figura 8, en la cual se puede observar la forma como se clasifican las palabras compuestas.

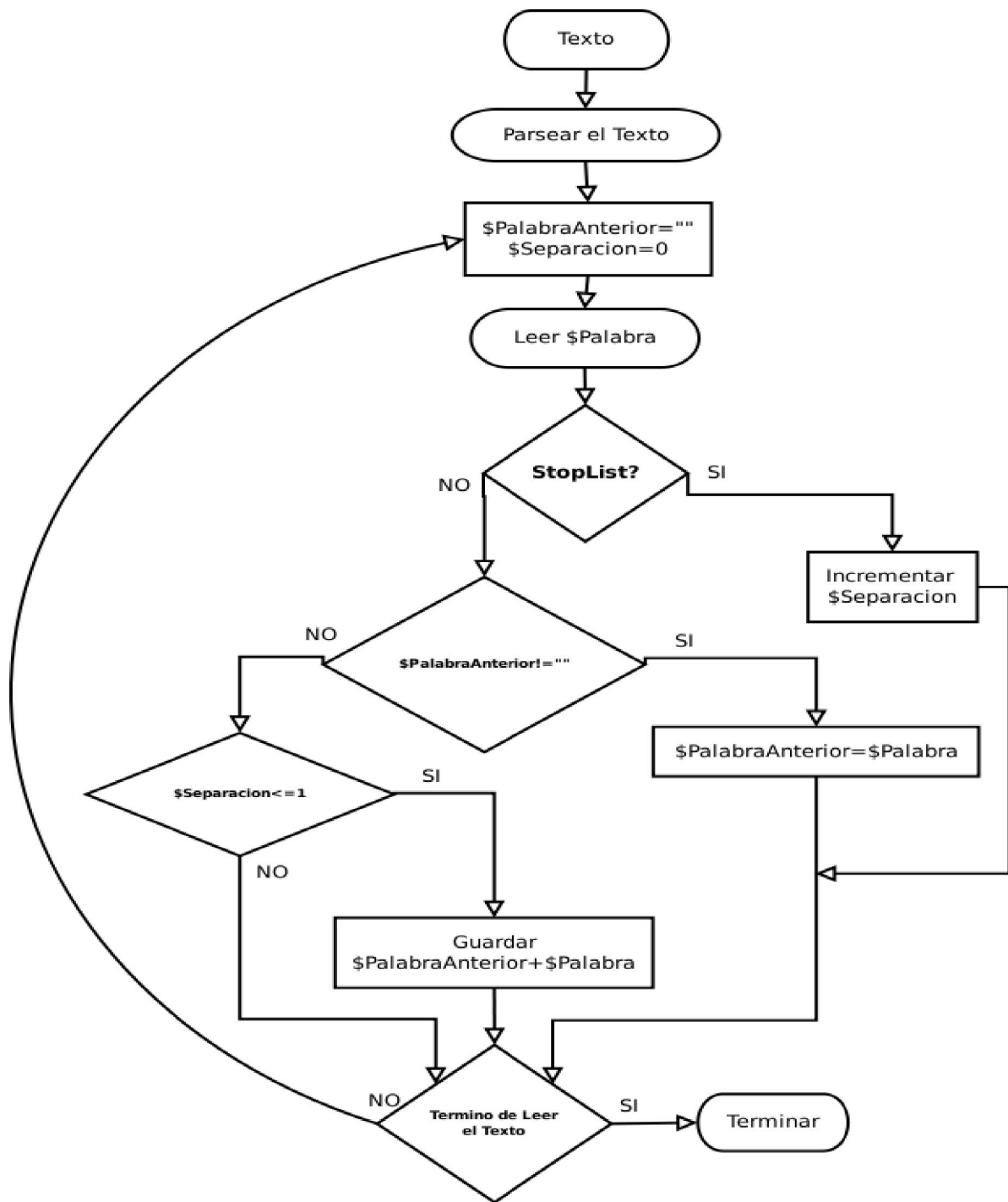


Figura 8: Diagrama de flujo del Agente de Memoria.

Este algoritmo muestra la forma de obtener las llamadas palabras compuestas.

Como se puede observar solo son palabras separadas por una palabra en la lista “StopList”. Si se desea palabras más largas es fácil modificarlo para realizar esta tarea.

2.4.4 Calcula_Tema

Aquí se busca la relación de cada palabra de forma individual para encontrar el Tema al cual pertenece o podría pertenecer. Una vez obtenida esa información, se utiliza la función Busca_Subtema para obtener la relación de subtemas a los que pertenece la palabra. El resultado es almacenado en un arreglo asociativo y devuelto por esta función.

2.4.5 Busca_Subtema

Busca y devuelve una lista compuesta por los subtemas relacionados por la palabra recibida , dicha lista lleva un orden por importancia de los subtemas relacionados, ordenados para poder obtener un mejor resultado de forma descendente.

2.4.6 Top_Subtema

La tarea de esta función es regresar los temas más relevantes de una palabra,

tomando como restricción regresar una tercera parte de los subtemas relacionados, tomando en cuenta que los temas están ordenados por importancia.

Toda la información consultada por estas funciones está contenida en una base de datos que puede ser editada, para mejoramiento del sistema.

2.4.7 StopList.pm

Este paquete solo cuenta con una función, “stoplist_search”, la cual busca una palabra en la lista de palabras no permitidas, o que no cuente con el tamaño mínimo para pasar. Se tiene definido un tamaño mínimo de 5 caracteres para que las palabras puedan pasar el filtro de “stoplist_search”, esto puede ser fácilmente removido o modificado. Si la palabra no es permitida según las reglas en esta función se devolverá un valor de “1” para avisar, de lo contrario se devuelve “0”.

La lista de palabras donde verifica “stoplist_search” se encuentra en una tabla de la base de datos del sistema, la cual es fácilmente editable. El funcionamiento de esta función se puede observar en la siguiente (Figura 9).

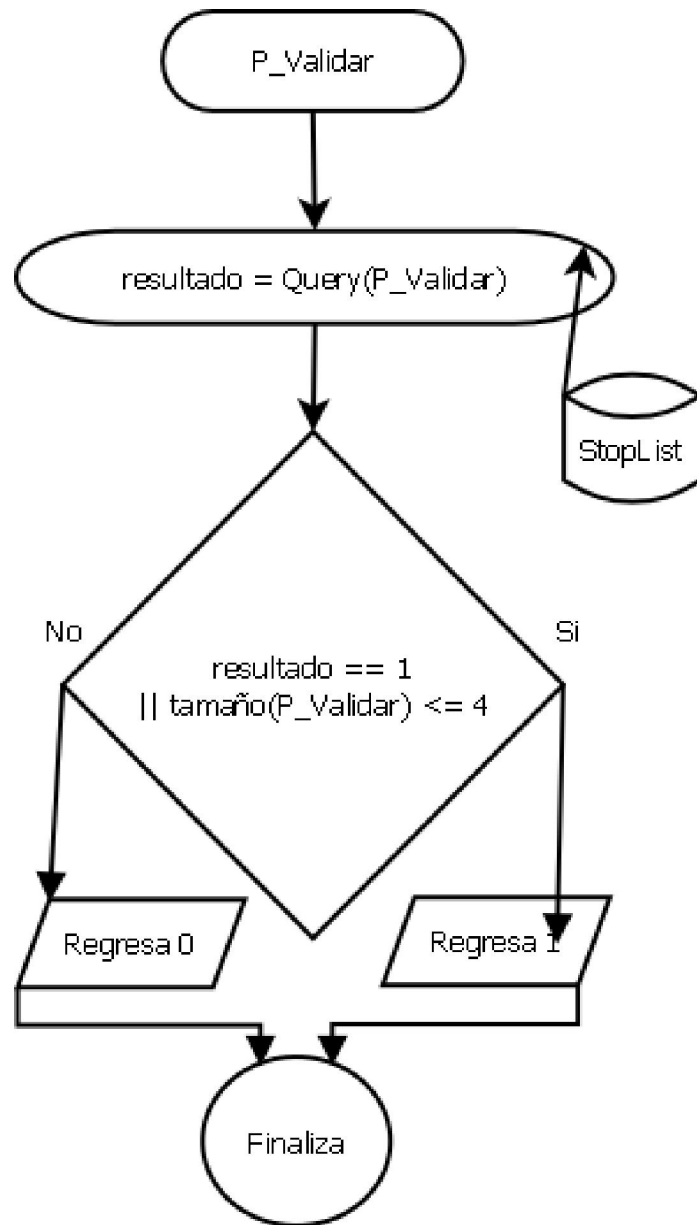


Figura 9: Funcionamiento del StopList.

2.4.8 Recomendador.pm

Este paquete realiza una búsqueda de los temas y subtemas de cada palabra.

Con esta información se realiza una relación con sus Recomendaciones y

Sugerencias que tengan relación con el tema de cada palabra. Una forma de verlo sería con el siguiente diagrama (Figura 10).

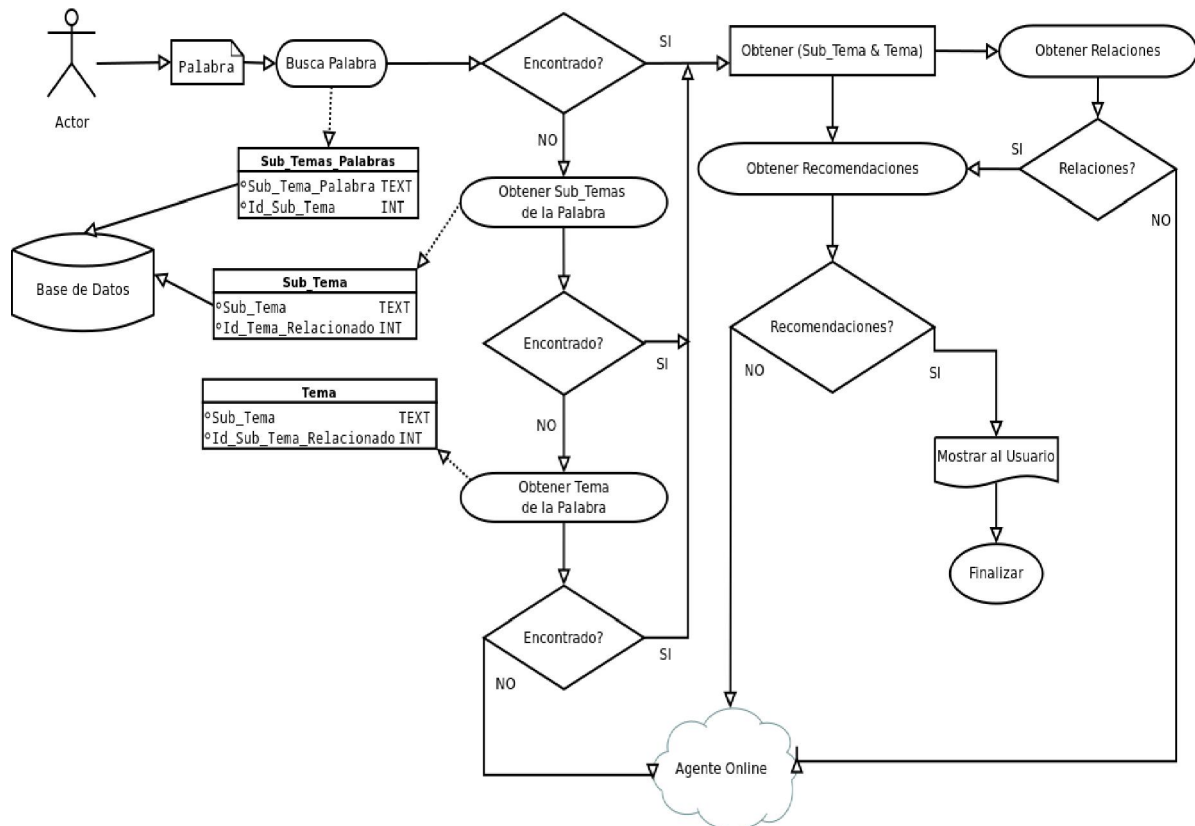


Figura 10: Diagrama de Flujo para la recomendación de una palabra.

Como se puede observar se realiza una búsqueda doble entre los temas a Recomendar y las Sugerencias para darle un mejor resultado al usuario. La idea de agregar el elemento “Agente Online” nació como parte de poder alimentar el Agente de Memoria preguntando a otros Agentes y así poder mejorar el análisis progresivamente. A continuación se describen las 2 funciones principales del modulo Recomendador.pm.

- **Sugerencia**

En base a los subtemas relacionados con el texto analizado, esta función genera una lista de sugerencias utilizando las funciones de “Recomendar_ (Texto, Web, Video, Imagen)”, estas son solicitadas en base a los temas que preceden o que se necesitan saber para comprender el tema analizado.

- **Recomendar_(Texto, Web, Video, Imagen)**

De una lista obtenida de subtemas, imprime en formato XML (XML es descrito en la sección 2.4.16 para su mejor comprensión) las recomendaciones de los subtemas, estas recomendaciones son obtenidas de una tabla que contiene todas las recomendaciones de cada subtema, como ejemplo del XML sería algo así:

```
<mensaje>  
<subtema>Nombre del Subtema</subtema>  
<tipo>Web,Video,Imagen</tipo>  
<texto>Liga sobre la recomendación</texto>  
</mensaje>
```

Las Recomendaciones y Sugerencias se basan en las tablas de relaciones de los temas en la base de datos, las cuales se pueden alimentar y modificar para un mejor resultado. Un ejemplo de esta relación entre temas se puede observar en

la siguiente figura (Figura 11).

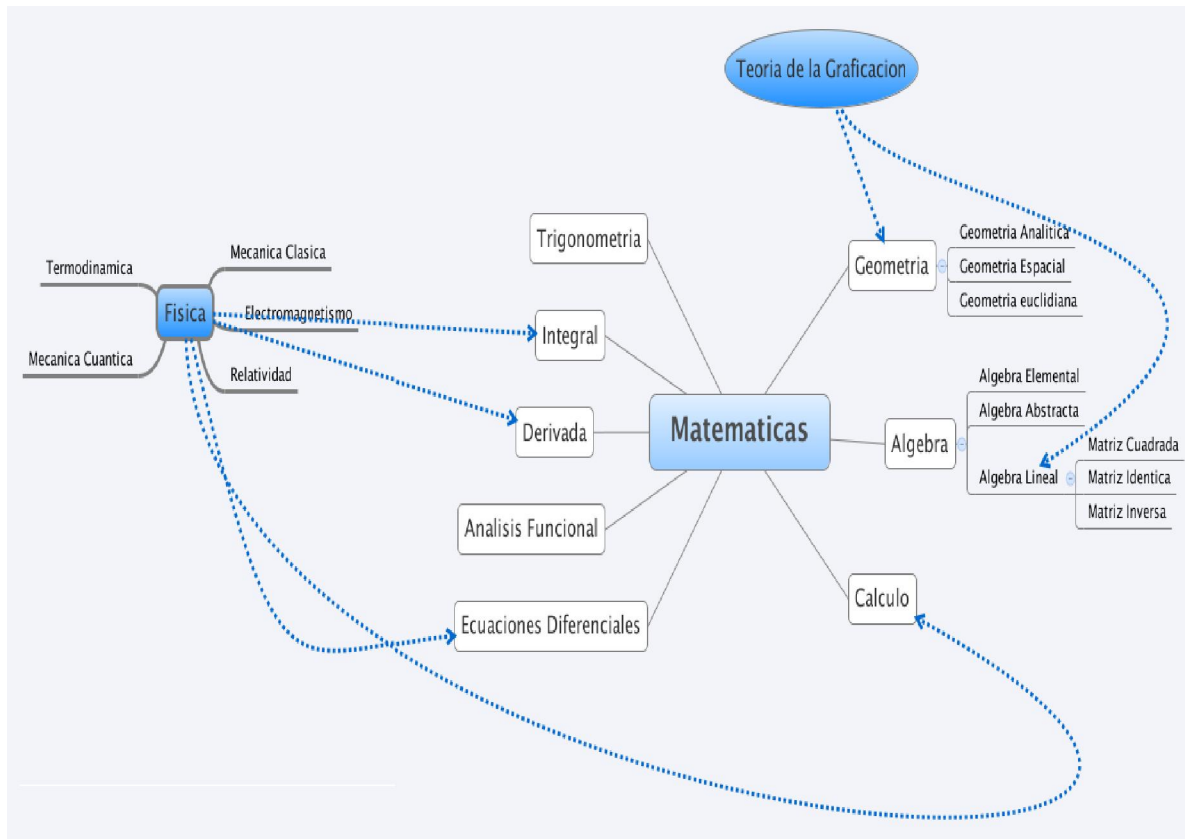


Figura 11: Diagrama de vínculos entre Temas y Subtemas.

Con esto el sistema aparte de dar una recomendación para el tema que se está analizando, puede dar sugerencias de temas que se necesitan dominar para entender de una mejor manera el tema. Esto es de mucha importancia, ya que existen personas estudiando un Tema específico sin dominar las bases, y eso lleva a complicaciones de comprensión y avance en su estudio.

2.4.9 ConectarDB.pm

Este es el paquete encargado de realizar la conexión a la base de datos MySQL, cuenta con una función llamada “connect” la cual nos regresará la conexión a la base de datos del proyecto en caso de ser posible, de lo contrario nos dará el error de conexión para ser analizado.

2.4.10 Agente.cgi

Este CGI hace de interfaz entre el agente y el cliente, hace utilización de las funciones de los paquetes y sus módulos antes mencionados. Es el encargado de recibir el texto el cual será procesado y devolver las respuestas dependiendo de la petición. El siguiente diagrama (Figura 12) muestra las acciones que pueden realizar y qué respuestas darían.

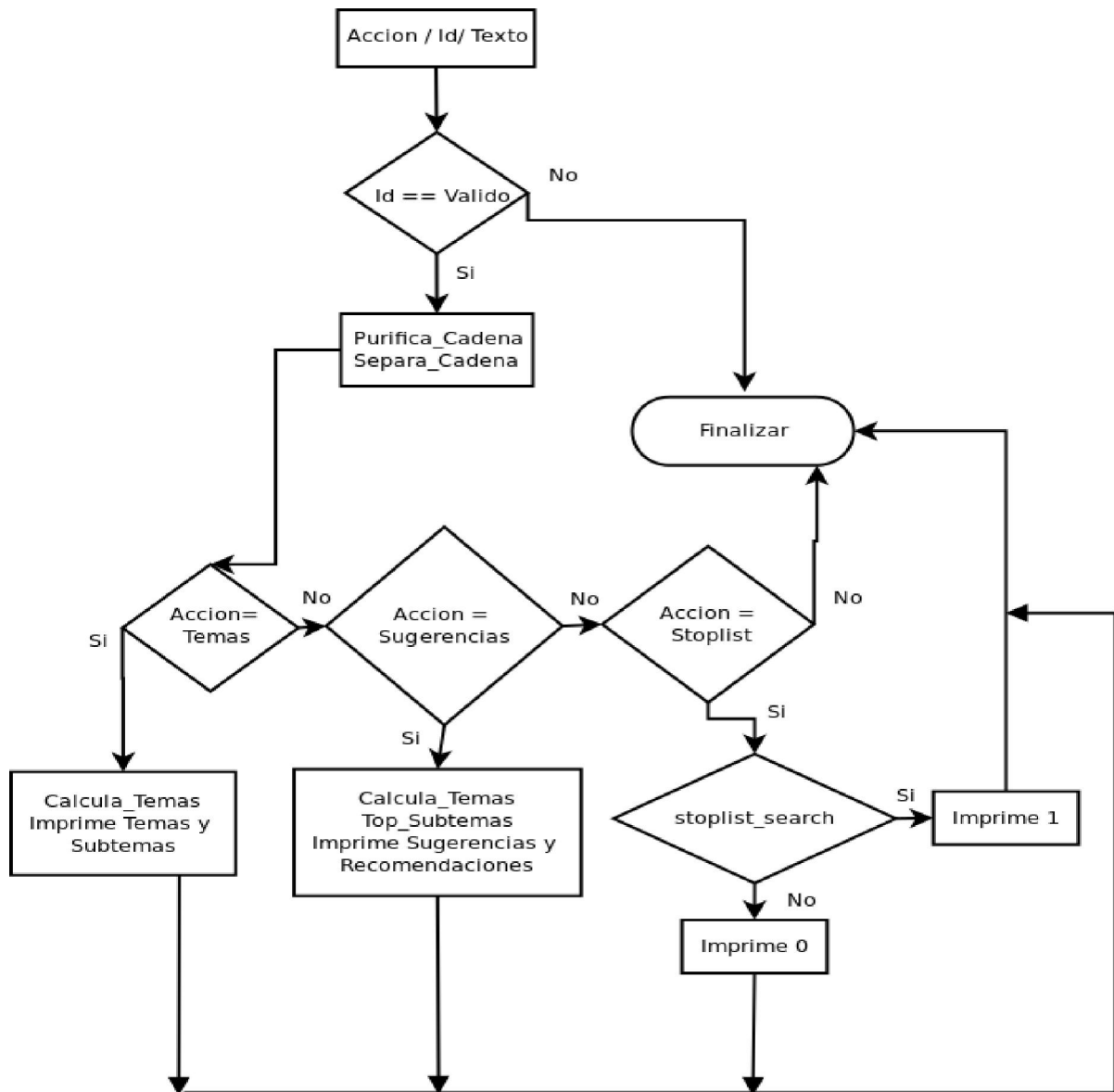


Figura 12: Diagrama de flujo del Agente usando todos sus módulos.

Como se muestra en el diagrama, el CGI puede realizar 3 tareas diferentes. Dependiendo de los parámetros utilizados al llamarlo será el tipo de respuesta obtenida.

Los parámetros permitidos son tres: Acción, ID, Texto. Estos parámetros se

describen a continuación:

1. Acción: Este parámetro nos sirve para elegir qué tipo de análisis va a realizar nuestro agente. Este agente realiza 3 tipos de análisis:

I. “Temas” devuelve un listado de los temas relacionados con el texto analizado.

II. “Sugerencias” al igual que el anterior pero son temas sugeridos sobre el texto.

III. “Stoplist” busca y regresa un valor positivo si la palabra enviada se encuentra en la lista de palabras no permitida para su análisis.

2. ID: Es utilizado para llevar un control del uso del agente, así para registrar quien utiliza el servicio y poder moderarlo.

3. Text: Por este parámetro se hace el envío del (texto/palabra) a la cual se le quiere aplicar el parámetro “Acción”.

La forma de utilizar y llamar al agente será descrita en el apartado “**Modelo de Comunicación con el Agente**” se explicará la forma de utilizarlo y el formato utilizado para responder.

2.4.11 Agente.pl

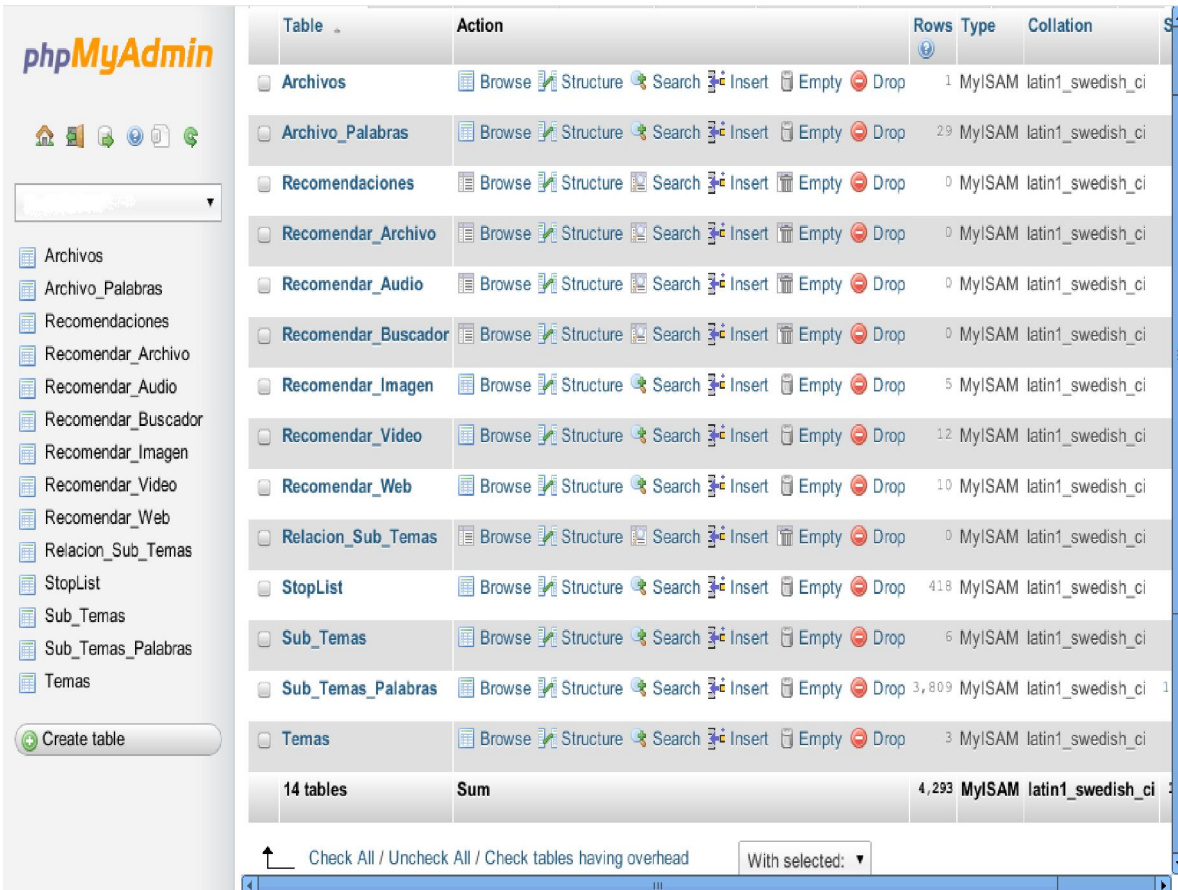
Una forma de probar el correcto funcionamiento de los paquetes y módulos es mediante Agente.pl el cual recibe como parámetro un archivo de texto y da como respuestas las recomendaciones. La ventaja de este Script es la utilización por medio de línea de comandos, así podremos comprobar el correcto funcionamiento de los módulos sin necesidad de modificar el CGI principal. La utilización de este Script es de la siguiente manera: “perl Agente.pl Archivo.txt”. Esta instrucción nos imprimirá las recomendaciones encontradas.

2.4.12 Carga-Tema.pl

Este Script nos ayuda para cargar palabras a un SubTema específico, esto para evitar estar ingresando una palabra a la vez en la base de datos. La forma de utilizarlo es “perl Carga-Tema.pl Archivo.txt SubTema”. Como se puede ver el sistema recibe 2 parámetros, el primero es el archivo que contiene las palabras relacionadas con el subtema, el segundo es el SubTema relacionado. Básicamente se leen las palabras separadas por espacio y se guardan en un arreglo. Después con el nombre del SubTema se busca su identificador y se agregan todas las palabras guardadas en el arreglo, para esto, el SubTema deberá existir previamente.

2.4.13 Base de datos

Al estar manejando una gran cantidad de datos y muchos están relacionados, se utilizó MySQL para este cumplir con este propósito, se crearon diversas tablas para tener control sobre las relaciones, en la siguiente (Figura 13) se muestran las Tablas de la base de datos:



The screenshot shows the phpMyAdmin interface with a table list. The table list has columns for Table, Action, Rows, Type, and Collation. The tables listed are: Archivos, Archivo_Palabras, Recomendaciones, Recomendar_Archivo, Recomendar_Audio, Recomendar_Buscador, Recomendar_Imagen, Recomendar_Video, Recomendar_Web, Relacion_Sub_Temas, StopList, Sub_Temas, and Sub_Temas_Palabras. A summary row at the bottom indicates 14 tables with a total of 4,293 rows.

Table	Action	Rows	Type	Collation
Archivos	Browse Structure Search Insert Empty Drop	1	MyISAM	latin1_swedish_ci
Archivo_Palabras	Browse Structure Search Insert Empty Drop	29	MyISAM	latin1_swedish_ci
Recomendaciones	Browse Structure Search Insert Empty Drop	0	MyISAM	latin1_swedish_ci
Recomendar_Archivo	Browse Structure Search Insert Empty Drop	0	MyISAM	latin1_swedish_ci
Recomendar_Audio	Browse Structure Search Insert Empty Drop	0	MyISAM	latin1_swedish_ci
Recomendar_Buscador	Browse Structure Search Insert Empty Drop	0	MyISAM	latin1_swedish_ci
Recomendar_Imagen	Browse Structure Search Insert Empty Drop	5	MyISAM	latin1_swedish_ci
Recomendar_Video	Browse Structure Search Insert Empty Drop	12	MyISAM	latin1_swedish_ci
Recomendar_Web	Browse Structure Search Insert Empty Drop	10	MyISAM	latin1_swedish_ci
Relacion_Sub_Temas	Browse Structure Search Insert Empty Drop	0	MyISAM	latin1_swedish_ci
StopList	Browse Structure Search Insert Empty Drop	418	MyISAM	latin1_swedish_ci
Sub_Temas	Browse Structure Search Insert Empty Drop	6	MyISAM	latin1_swedish_ci
Sub_Temas_Palabras	Browse Structure Search Insert Empty Drop	3,809	MyISAM	latin1_swedish_ci
Temas	Browse Structure Search Insert Empty Drop	3	MyISAM	latin1_swedish_ci
14 tables	Sum	4,293	MyISAM	latin1_swedish_ci

Figura 13: Panel de administración de las tablas del sistema.

Se creó una tabla para cada tipo de recomendación ya sea Imagen, Video, Web, Archivo, Audio, Buscador específico. La tabla de archivos se utiliza para tener un control de archivos indexados que se encuentran en el servidor, los Temas y Subtemas están separados en 3 tablas para poder tener un mejor indexado y control sobre ellos. En la siguiente Imagen (Figura 14) se muestra el contenido de dichas tablas.

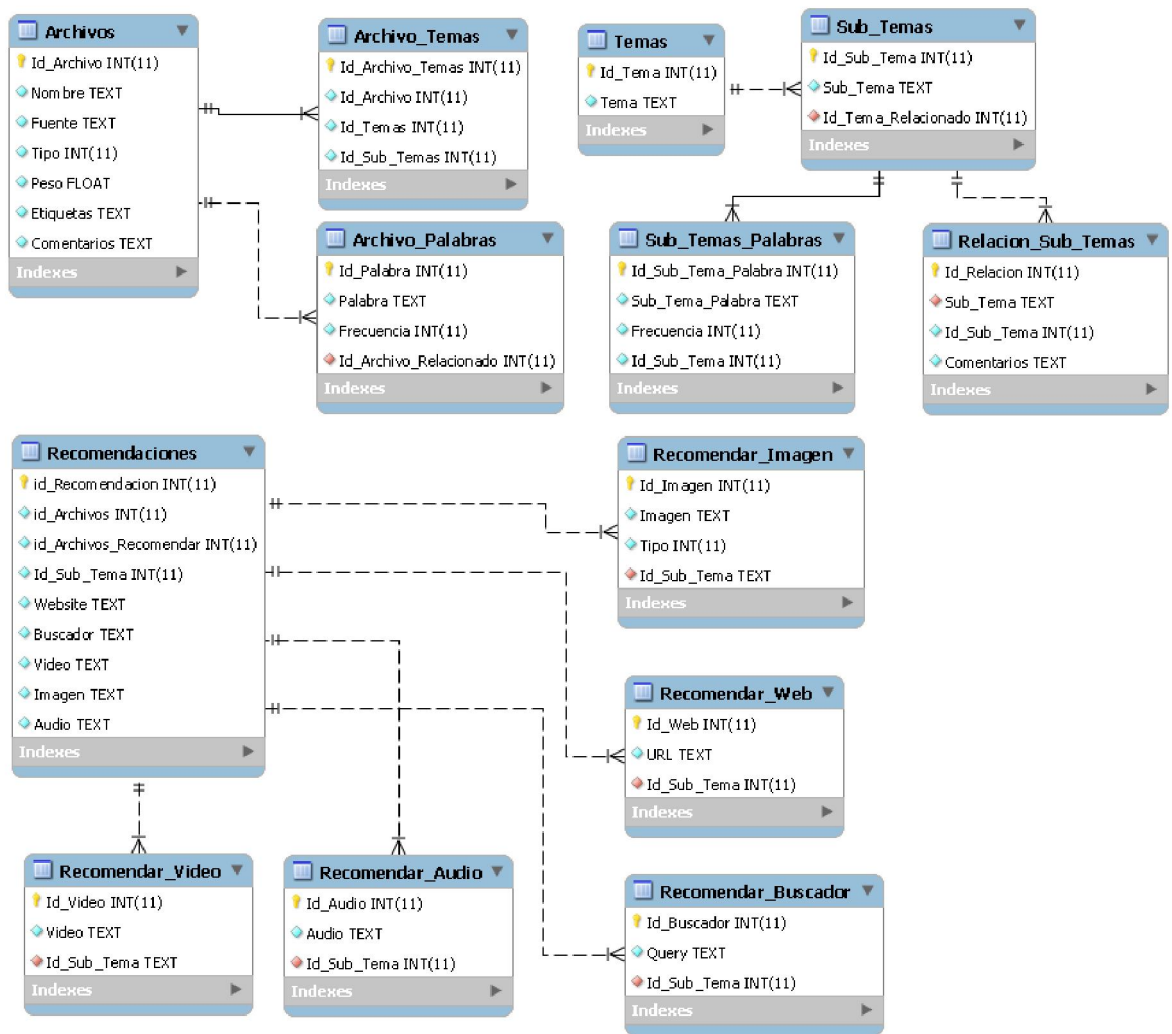


Figura 14: Tablas y su contenido.

Como se puede observar, para indexar el contenido y los archivos en el servidor se utilizan las tablas Archivos, Archivos_Palabras, Archivo_Temas, Recomendaciones. Estas tablas guardan la información básica del archivo así como sus palabras extraídas y los temas y subtemas encontrados que tienen relación con él.

En las tablas de Recomendar_Imagen, Recomendar_Audio, Recomendar_Video, Recomendar_Web, Recomendar_Buscador se guardan las recomendaciones de diversos tipos. En todas ellas se guarda la dirección web de la recomendación. La tabla de Recomendar_Buscador lo que proporciona es la consulta (query) para realizar la búsqueda en un buscador externo.

Para tener la relación de palabras y subtemas se utilizan las tablas Temas, Sub_Temas, Sub_Temas_Palabras, Relacion_Sub_Temas.

2.4.14 Protocolo de Comunicación con el Agente

El protocolo elegido para la transferencia de la información fue REST. Este cumple con los requisitos de ser independiente del lenguaje y plataforma en la que se creen los clientes, además es escalable, fácil de aprender y actualizar. REST se basa en un sistema de envío de mensajes, se puede trabajar de forma Síncrona y Asíncrona. Utiliza el protocolo HTTP, en particular el comando GET para su comunicación, de esta forma no requiere de grandes configuraciones y/o modificaciones del lado del servidor.

En el punto 2.4.15 se detallan los componentes para utilizar REST en este proyecto.

2.4.15 Modelo de Comunicación con el Agente

La arquitectura de utilización, al ser por paso de mensajes al servidor, es sencilla. De igual manera es la forma de recibir la respuesta. La comunicación se basa en un script del lado del servidor. El script utilizado es Agente.cgi, el cual recibe 3 parámetros para su funcionamiento, que son Orden, ID y Texto. La manera de utilizar los parámetros que se utilizan en el agente es la siguiente.

Orden: Este parámetro nos permite elegir entre 3 diferentes acciones a realizar, se detallan a continuación:

- Sugerencias: Obtener la lista de Sugerencias y Recomendaciones sobre el parámetro Texto.
- Temas: Obtener temas relacionados con el Texto.
- Stoplist: Recibe una sola palabra en el Texto, busca en la tabla de “Palabras no válidas” Stoplist y responde si pertenece o no a ella.

Usuario: Es un registro de usuarios válidos, para llevar una estadística de quienes usan el servicio, además una medida para evitar que otros usen el sistema. Para este proyecto se agrega al servidor la dirección IP del usuario que está realizando las pruebas para mayor seguridad y así evitar usuarios no

registrados o no deseados.

Texto: Texto o palabra a analizar por el Agente de Memoria, siguiendo el parámetro Orden.

Para realizar pruebas y ver el funcionamiento del Agente, se utiliza esta liga:

<http://ia.ens.uabc.mx/cgi-bin/Agente.cgi?Orden&Usuario&Texto>. La forma en que funciona esta modelo de comunicacion se muestra en la siguiente (Figura 15).

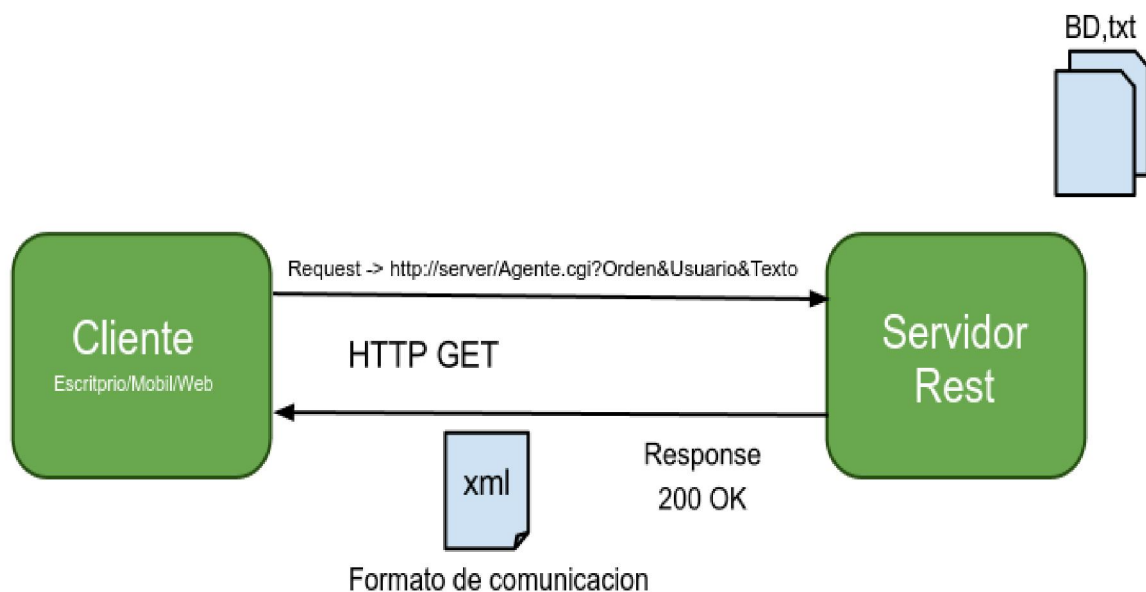


Figura 15: Diagrama que muestra la forma de comunicarse con el Agente.

El cliente, que puede ser una App en Escritorio, Dispositivo móvil o en la WEB, realiza la petición al servidor por medio del protocolo de comunicación

HTTP, el protocolo utilizado para páginas web. El servidor responderá por medio de texto en formato XML para su fácil manipulación del lado del Cliente. Esta arquitectura es la más cómoda para programadores y para administradores de servidores, ya que no requiere de conocimientos técnicos o complejos para llevarlo a cabo.

2.4.16 Formato

El formato de comunicación utilizado es XML, por ser fácil de implementar y no requiere de configuraciones del programador ni del servidor, el formato del XML utilizado en el Agente es el siguiente:

```
<?xmlversión='1.0'?>
<mensajes>
<mensaje>
<subtema>Subtema</subtema>
<tipo>Web</tipo>
<texto>www.paginadesubtema.com</texto>
</mensaje>
```

Como se puede observar el paquete XML que se maneja para responder es sencillo. Cuenta con la etiqueta <mensajes> que engloba todo el paquete. Cada respuesta se deposita en una etiqueta <mensaje>, la cual está compuesta

por <subtema>, <tipo>, <texto>, que son los datos de respuesta por parte del Agente.

2.5 Editor de Texto (WEB)

Se desarrolló un editor de texto en línea para realizar pruebas reales al Agente.

El editor consta de 4 secciones:

- Cuadro de texto. Es donde el usuario pondrá el texto a analizar.
- Temas. Se muestra una lista de temas que tiene alguna relación con el texto analizado.
- Palabras. Muestra las palabras individuales y las palabras compuestas.
- Recomendaciones/Sugerencias. Aquí se despliega el resultado devuelto por el Agente.

Para una mejor apreciación, en la siguiente imagen (Figura 16) se puede ver un ejemplo de la interfaz y las secciones del Editor:

Editor Remembrance Agent

[Cargar Ejemplo] Ejemplo 2

Temas

Calculo Diferencial
Graficacion
Geometria
Electromagnetismo
Algebra
Mecanica Cuantica

Palabras

(1) vectorial
(2) soluciAn distinta
(1) semejantes
(1) endomorfismo
(2) sistema ecuaciones
(1) desarrollo
(1) vectorialde
(2) endomorfismo definido
(2) espacio vectorialde
(1) espacio
(1) sistema
(1) matriz
(1) orden
(1) matrices
(1) ecuaciones
(2) polinomio matrices

Dos matrices cuadradas de orden n , A y B , son semejantes si existe una matriz S Sea vectorial un endomorfismo definido sobre un espacio vectorialde dimensiAⁿ n Este sistema de ecuaciones tendrA; soluciAⁿ distinta de latrivial si el determinante de ... El desarrollo de trivial determinanteda lugar a un polinomio de matrices

Procesa Agente

Recomendaciones/Sugerencias:

Web [Calculo Diferencial]: http://es.wikipedia.org/wiki/C%C3%A1culo_diferencial
Web [Calculo Diferencial]: <http://dieumsh.qfb.umich.mx/DIFERENCIAL/diferencial.htm>
Video [Calculo Diferencial]: http://www.youtube.com/watch?v=iT_xJ7OPRZo
Video [Calculo Diferencial]: <http://www.youtube.com/watch?v=jxksTUbLORU>
Imagen [Calculo Diferencial]: uabc.jpg
Web [Electromagnetismo]: http://www.asifunciona.com/electrotecnia/ke_electromag/ke_electromag_1.htm
Web [Electromagnetismo]: <http://es.wikipedia.org/wiki/Electromagnetismo>
Video [Electromagnetismo]: <http://www.youtube.com/watch?v=7s7Cd1ZjO2M>

Copyright ©

Figura 16: Editor de texto para el Agente.

Para dar inicio al análisis se utiliza el botón “Procesa Agente”, se cuenta con dos ejemplos de textos para realizar las pruebas “Cargar Ejemplo, Ejemplo 2”.

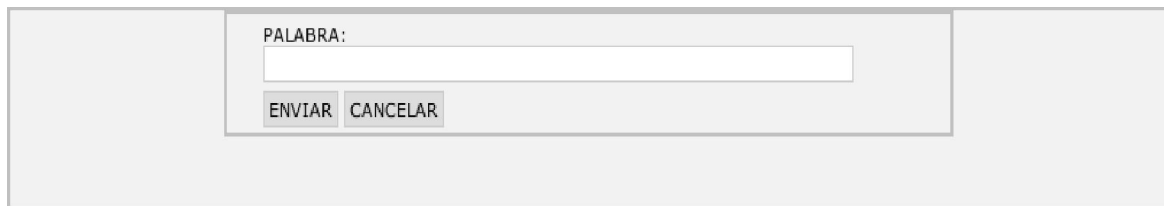
2.6 Editores de contenido

Se crearon un par de editores de apoyo para alimentar y/o editar la base de datos para las recomendaciones/sugerencias.

2.6.1 Editor StopList

Para facilitar la alimentación de palabras a la tabla StopList, se creó un editor de tal manera que se evite el uso directo de la tabla en la base de datos . A

continuación se muestra la imagen del módulo (Figura 17):



The image shows a light gray rectangular container. Inside, there is a smaller gray box. At the top of this box is the label "PALABRA:". Below the label is a white rectangular input field. Underneath the input field are two buttons: "ENVIAR" on the left and "CANCELAR" on the right, both with a light gray background and dark text.

Figura 17: Modulo agregar palabras a StopList.

Para borrar o editar dichas palabras se utiliza el módulo del editor de palabras en lista, como se muestra en la siguiente (Figura 18):

ID PALABRA	PALABRA		
423	LUGAR		
422	SIMON		
421	INGESU		
416	HAGAN		
415	DADAS		
413	DEPENDE		
412	DEMAS		
411	DICHOS		
410	CUALES		
409	VECES		
408	DICIENDO		
407	SERAN		
406	PASAN		
405	POSEE		
404	SUAVE		
403	HACERSE		
402	SEGUIR		
401	LLEVAN		

Figura 18: Editor de contenido de StopList.

Como se puede ver en el editor, proporciona opción de borrar la palabra o editarla, cuenta con una parte de búsqueda por palabra o ID de la palabra.

2.6.2 Editor de Palabras en Subtemas

Con este editor se pueden modificar las palabras relacionadas con los

subtemas. También se puede modificar la misma palabra, frecuencia, Id-subtema, subtema relacionado, y su Valor (Figura 19).

MENU DE SUBTEMAS

ID PALABRA	PALABRA SUBTEMA	FRECUENCIA	ID-SUBTEMA	VALOR	
4739	COMPUTADORA	22	11	50	 
4737	GRAFICA	10	11	9	 
5386	MATRIZ VECTORIAL	8	1	9	 
5388	MATRIZ VECTORIAL	4	3	7	 
5389	MATRIZ VECTORIAL	5	11	7	 
3079	VECTORIAL	1	13	6	 
5387	MATRIZ VECTORIAL	3	2	6	 
1992	TRIVIAL	1	1	5	 
5390	MATRIZ VECTORIAL	2	13	4	 
3323	MATRICES	1	13	2	 
1854	MATRIZ	1	1	1	 
1581	ESTUDIO	3	1	0	 
1582	FUNCION	70	1	0	 

Figura 19: Editor Subtemas.

Se debe tener cuidado al editar la frecuencia, ya que es un valor muy delicado debido a que afecta directamente los resultados del algoritmo. Este valor denota la importancia de esta palabra en el subtema al que pertenece.

2.6.3 Indexar un Archivo

Para indexar archivos “imágenes, textos y más” se creó un el siguiente módulo que se muestra en la (Figura 20).

Elije Tema para el archivo =>

Calculo Diferencial-1	
Algebra-2	
Geometria-3	
Mecanica Cuantica-12	
Electromagnetismo-13	
Graficacion-11	

Añadir a la segunda lista Quitar item de la segunda lista

Comentario:

Puntos 1 2 3 4 5 6 7 8 9 10

Elegir un nuevo archivo: No file chosen

Figura 20: Modulo indexar un archivo.

Este cuenta con la lista de Subtemas que podemos elegir para relacionar con el archivo a anexar, la sección de Puntos es para darle una valoración al archivo sobre su importancia en cuanto a los demás, después ya solo queda elegir el archivo y presionar Subir.

2.7 Requerimientos para instalación

Para poner en marcha el Agente de Memoria, se tiene los siguientes requerimientos para la instalación:

- Sistema Operativo con soporte para mantener un servidor HTTP.
- Software para la instalación de un servidor HTTP, con soporte del lenguaje PHP y PERL.
- MySQL server para las base de datos y sus módulos para trabajar con el lenguaje PERL.
- Espacio suficiente para el contenido de la base de datos y archivos respaldados en el servidor por el agente de memoria.

Este es libre de la plataforma que se quiera utilizar, mientras cumpla con los requerimientos. Para esta investigación se utilizo la siguiente configuración

- Linux (CENTOS) con kernel “2.6.32-279.14.1.el6.x86_64.
- PERL en la versión “5.10.1”.
- MySQL en la versión “14.14”
- PHP la versión “5.3.3”.

Una vez teniendo un servidor con estas características es necesario realizar

algunas pruebas antes de utilizar el Agente, primero verificar que el servidor soporte ejecutar CGI de Perl, una forma es con el siguiente código:

```
#!/usr/bin/perl
print "Content-Type: text/html\n\n";
print "<?xml version='1.0'?>";
print "Hello, world!\n";
```

Una vez teniendo el cgi en el folder correspondiente del servidor, se pasa a realizar una simple prueba como esta:

<http://127.0.0.1/cgi-bin/prueba.cgi>

**Teniendo en cuenta que el folder para scripts de Perl sea cgi-bin en algunos servidores puede ser otro o uno mismo lo puede configurar*

Esto nos debería dar como salida en la pantalla del navegador:

Hello, world!

Una vez comprobado esto ya se puede hacer uso del Agente de Memoria, así como sus editores complementarios y Editor de Prueba.

3 Pruebas

Para realizar las pruebas, se agregaron 3 temas y sus respectivos subtemas, así como una considerable cantidad de palabras relacionadas con estos subtemas.

A continuación se muestran la información utilizada para esta prueba:

Matemáticas (Calculo Diferencial, Algebra, Geometría)

- Ecuaciones
- Matriz vectorial
- Función
- Calculo
- Calculo diferencial
- Matriz

Física (Mecánica Cuántica, Electromagnetismo)

- Ecuaciones
- Vectorial
- Matriz vectorial
- Matrices

Gráficos (Graficacion)

- Computadora
- Grafica
- Matriz vectorial

Cada palabra relacionada cuenta con un porcentaje de importancia. Se realizó la relación entre temas y subtemas, esto con el fin de poder enriquecer el análisis de los temas. Ya contando con este las pruebas se basan en la cantidad de palabras más frecuentes en el texto, además de las palabras con más porcentaje en los temas, esto para evitar un análisis solo por conteo de palabras. Con este método se puede obtener un mejor filtro de los temas encontrados en el texto analizado, así si un texto tiene mucho contenido repetido pero cuenta con palabras bases y de relevancia en algunos temas podrá destacar en los resultados.

Se realizaron 3 tipos de pruebas que son el mejor de los casos, la media y el peor de los casos.

- **Mejor de los casos:** En este caso el texto a analizar cuenta con una cantidad mayor de palabras sobre un tema contenido en la base de datos, esto da como resultado una fácil identificación de las recomendaciones y sugerencias sobre el texto analizado.

- **La Media:** En este caso los subtemas encontrados en el texto analizado contienen el mismo porcentaje de importancia, esto podría dificultar encontrar la correcta sugerencias/recomendaciones del texto, esto se soluciona tomando en cuenta que cada palabra relacionada cuenta con un porcentaje de importancia sobre el subtema, se realiza una simple sumatoria de estos porcentajes y así se evalúa cual es el tema que sobre sale del texto, en debido caso de que esto llevara a igual el orden de relevancia de los subtemas a sugerir, se toman la misma cantidad de sugerencias/recomendaciones para todos los que sean iguales en importancia. Tomando un texto del cual se obtiene 3 palabras a analizar, tenemos la siguiente tabla (Figura 21).

Subtemas	Palabras Relacionadas	Valor de las Palabras	Suma Total	Posición
Graficacion	P1, P2	10,20	30	2
Geometría	P1, P2, P3	5,15,10	30	1
Calculo	P1	15	15	3

Figura 21: Tabla de análisis para "La media"

Donde cada “Palabra Relacionadas” tiene un valor de importancia en cada subtema, el valor de cada palabra es dado por un experto en el tema, la posición está dada por la suma del total de los valores de las palabras, en caso de valores repetidos se toma el que más palabras contenga.

- **Peor de los casos:** Aquí se presentaría cuando los porcentajes de importancia de las palabras no esté equilibrado y el Agente de Memoria de

como resultado un subtema incorrecto , o el caso de que 2 subtemas cuenten con las misma palabras relacionadas y porcentajes diferentes, esto puede prestar a confundir el resultado del agente, estos casos se pueden dar cuando una palabra es repetida en varias ocasiones , provocando que los porcentajes en la sumatoria suban, por eso el sistema cuenta con una prevención al encontrar este tipo de problemas. Se podría tomar como un peor de los casos a la ausencia de subtemas encontrados en el texto a analizar, pero se respuesta es la ausencia de sugerencias/recomendaciones y pediría al usuario que aporte alguna para alimentar al sistema y aportar algo sobre ese tema.

Para realizar estas pruebas se cuenta con un Script para usarlo desde terminal del sistema o invocarlo como un cgi directamente desde un navegador.

El script para la consola se llama Agente.pl el cual recibe como parámetro un archivo para analizar, este devuelve la impresión del XML como resultado.

Se realizaron pruebas con textos de tamaños de 1 hasta más de 1000 palabras y el Agente se comportó funcional en todos los casos.

Ahora nos queda probar el servicio REST de nuestro cgi, esto lo hacemos

desde el navegador de nuestra preferencia, esto lo realizamos accediendo al servidor y a al folder contenedor del cgi. La ruta para las pruebas realizadas que se utilizo fue la siguiente: <http://ia.ens.uabc.mx/cgi-bin/Agente.cgi>, los cambios pueden ser depende la configuración de tu servidor, ya teniendo esto nos saldrá una pantalla en blanco, señal de que todo marcha bien, ahora solo queda probar los parámetros que permite el Agente.cgi “Temas, Sugerencias, Filtro, StopList”.

Para realizar las siguientes pruebas se dejó el usuario “User” como el predeterminado para realizar pruebas al sistema, ahora pasamos a realizar pruebas individuales a cada parámetro.

Sugerencias: <http://ia.ens.uabc.mx/cgi-bin/Agente.cgi?Sugerencias&User&Texto> estos nos devuelve una lista de sugerencias de haberlas encontrado en la base de datos y analizadas con el algoritmos de sugerencias.

Temas: <http://ia.ens.uabc.mx/cgi-bin/Agente.cgi?Temas&User&Texto> esto nos devuelve un listado de los temas a los cuales pertenecen las palabras analizadas.

Filtro: <http://ia.ens.uabc.mx/cgi-bin/Agente.cgi?Filtro&User&Texto> este analiza el texto y nos devuelve las palabras que serían analizadas por el sistema con los parámetros “Temas y Sugerencias”

Stoplist: <http://ia.ens.uabc.mx/cgi-bin/Agente.cgi?StopList&User&Palabra> esto nos responderá un 1 si la palabra se encuentra en tabla de la palabras no permitidas y un 0 si no lo está.

Las pruebas realizadas aquí se basaron en textos extraídos de libros sobre los temas ingresados en la base de datos, exámenes sobre esos temas y algunos apuntes diversos relacionados con el área de ingeniería en donde estos temas son de uso común.

3.1 Resultados de las pruebas

Los resultados de esta investigación muestran que para obtener buenos resultados el Agente, no es necesario que el texto a analizar contenga una gran cantidad de subtemas, si el tema contiene las palabras claves del tema su identificación será óptima, de los textos analizados se logró obtener resultados satisfactorios.

Tomando en cuenta la base de datos con la que se cuenta, los resultados

obtenidos sobre las sugerencias/recomendaciones sobre estos temas son 90% acertados.

Tomando que el trabajo más cercano a este, solo analiza pequeñas cantidades de información y relaciones, se puede llegar a la conclusión que el sistema aquí desarrollado cumple con esas expectativas y las supera.

Se sugiere alimentar más el sistema así se expandirá más el conocimiento del Agente y se evitara el encontrar temas repetidos o no muy acertados. La probabilidad de que los subtemas encontrados sean repetidos es muy grande, para eso se toman una estadística que se muestra en la tabla a continuación (Figura 22).

Palabras	Subtemas	Temas Repetidos 35%	Sin Repetir
1	3	1.05	1.95
2	6	2.1	3.9
3	9	3.15	5.85
4	12	4.2	7.8
5	15	5.25	9.75
6	18	6.3	11.7
7	21	7.35	13.65
8	24	8.4	15.6
9	27	9.45	17.55

Figura 22: Tabla de Estadística.

En base a esta estadística se puede obtener mejor la calidad de subtemas para repetir sugerencias/recomendaciones al usuario.

La calidad de respuesta en las sugerencias, depende mucho de la pureza en la

relación de las palabras y sus temas, se debe tomar en cuenta las palabras no tomadas por el algoritmo, esto ayuda al análisis a la hora de tomar la decisión de los temas a recomendar.

Los resultados obtenidos con en textos grandes fueron satisfactorios, con una precisión de un 80% a 85% en la obtención de los subtemas del texto analizado, esto debido a que la alimentación de la base de datos se basó en las breve descripciones o palabras claves de los subtemas.

En textos cortos se encontró un porcentaje un entre 60% a 70%, esto dependiendo de la cantidad de las palabras analizadas, cuando se tiene de 1 a 3 palabras para analizar se dificulta discernir el tema del texto.

El Agente es funcional en las pruebas realizadas, al utilizar un lenguaje de programación apto para el procesamiento de grandes texto, este favoreció mucho los tiempos de procesamiento del texto. Sería necesario alimentar más la base de datos por medio de un módulo exclusivo para que personas expertas en los temas elija las palabras de los subtemas y su peso en ella.

4 Conclusiones, Aportaciones y Trabajo a Futuro

Al finalizar la tesis, se cumplió con los objetivos marcados por la investigación, se diseño y creó un Agente de Memoria acorde al objetivo de apoyar la educación individual, se desarrollo bajo tecnología escalable y de fácil mantenimiento. Se crearon editores para facilitar la alimentación de la base de datos del Agente de Memoria. Se cumplió la creación de un Editor WEB para realizar las pruebas necesarias al Agente de Memoria, todo esto utilizando herramientas de software libre y de fácil configuración, esto para evitar problemas a la difusión del Agente de Memoria aquí creado.

Tenemos como aportación la base de un Agente de Memoria para ser modificado para otros fines de investigación, una estructura de base de datos para aprovecharla, así como una guía y código documentado para su fácil acceso.

Como trabajo a futuro sería necesaria la creación de aplicaciones en dispositivos personales, como lo son teléfonos móviles, Tablet y nuevas tecnologías para el acceso a la información. Fusionar con el agente un algoritmo existente para el reconocimiento de textos como los OCR y la alimentación por parte de ellos [12], crear aplicaciones más inteligentes que ayudan al sistema a aprender más sobre los textos analizados [18], ya sea

editando el peso de las palabras en los subtemas o alimentado al sistema con información previa al análisis. Actualmente se vive el nacimiento de los buscadores semánticos [19] estos podrían aportar una gran ayuda al Agente de Memoria, utilizar un sistema de reconocimiento de imágenes y compararla con la base de datos de imágenes sobre los temas, así contar con más variables a la hora realizar el análisis. Utilizar la Realidad Aumentada como medio para exponer la información sería lo más adecuado, tomar en cuenta el perfil del estudiante así como sus aportaciones a la hora de aceptar o rechazar recomendaciones o sugerencias.

Esta considerado que este trabajo podría tener un impacto positivo en la forma en que se maneja y acceden a la información las personas actualmente, pudiendo así aportar a la comunidad estudiantil y académica una nueva forma de manejar y procesar la información.

Se concluye que el Agente creado satisface las necesidades planteadas al principio de la investigación.

5 APÉNDICE A: Guía Instalación del Sistema

Para instalar el Agente es necesario tener un sistema con los requerimientos, una vez teniendo esto se debe seguir estos sencillos pasos:

- Los archivos pertenecientes al Agente “.cgi, .pm y .pl” deberán estar en una sola carpeta, la cual es la que está configurada para poder ejecutar los CGI en el servidor.
- Ahora utilizando MySQL ya sea por medio de la línea de comandos que el mismo proporciona o algún manejador gráfico “phpMyAdmin” importar la base de datos “Conocimiento.sql”.
- Editar el archivo “ConectarDB.pm” para editar los campos para realizar la conexión a la base de datos por medio del Agente.
- Una vez realizado esto, realizar una prueba con “Agente.pl”.
- Si todo marcha bien el Editor para realizar las pruebas por medio del navegador, deberá ser copiado a un folder el cual sea accesible por la web.
- Se edita el archivo “agente-editor.php” para agregar la dirección web donde se encontrará el Agente.

Ya teniendo esto configurado ya tenemos el agente instalado y listo para usarse.

6 Referencias

[1] **Dr. Tevni Grajales G. (2002)**. “Hábitos de estudio de estudiantes universitarios. Centro de Investigación Educativa”, (1997), 1-22.

[2] **P. Esteban, J. Restrepo, H. Trefftz, J.E. Jaramillo, and N. Alvarez**, “La realidad aumentada: un espacio para la comprensión de conceptos del cálculo en varias variables.”

[3] **Basogain, X., Olabe, M., Espinosa, K., & Olabe, C. R. J. C. (n.d.)**. “Realidad Aumentada en la Educación: una tecnología emergente”.

[4] **Rhodes, B. J. (1997)**. “The wearable remembrance agent: a system for augmented memory. Digest of Papers. First International Symposium on Wearable Computers”, 123-128. IEEE Comput. Soc. doi: 10.1109/ISWC.1997.629928.

[5] **Jauhiainen & Eskola: Ryhmäilmiö**. “Entorno de aprendizaje social”, [Enlace: http://cibernarium.tamk.fi/havainnolistaminen_es/social_environment2.htm] , [Consultada: 2012]

[6] **Starner, T., Mann, S., Rhodes, B., Levine, J., Healey, J., Kirsch, D., et al. (1997)**. “Augmented reality through wearable computing. Presence: Teleoperators and Virtual Environments”, 6(4), 386–398. Citeseer. [Enlace: <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.26.2886&rep=rep1&type=pdf>], [Consultada: 2010].

[7] **N. Alvarez, J. Jaramillo, J. Restrepo, H. Trefftz, P. Esteban**. “Augmented Reality for Teaching Multi-Variate Calculus. En A. M. Vilas, J.A. M. Gonzalez, J. M. Gonzalez (Eds) Advances in Technology-Based Education: Toward a Knowledge-Based Society.

[8] **Erick Klopfer**, “MIT Handheld Augmented Reality Simulations”, [Enlace: <http://education.mit.edu/ar/>], [Consultada: 2010].

[9] **Google Inc.**, Google Goggles, [Enlace: <http://www.google.com/mobile/goggles/#text>], [Consultada: 2010].

[10] **Alchemysnap** = OCR + Camera Phone Search , [Enlace: <http://www.alchemysnap.com/>], [Consultada: 2010].

[11] **Hans Peter Luhn**, [Enlace: http://es.wikipedia.org/wiki/Hans_Peter_Luhn], [Consultada: 2012]

[12] **Hahn, S., & Kim, J.** (1999). “A study on utilizing OCR technology in building text database. *Proceedings. Tenth International Workshop on Database and Expert Systems Applications*” .DEXA 99, 582–586. doi:10.1109/DEXA.1999.795250

[13] <http://perldoc.perl.org/> , “Documentación en español del lenguaje Perl”, [Consultada 2010]

[14] **W3**, http://es.wikipedia.org/wiki/Representational_State_Transfer, “Definición de la arquitectura REST”, [Consultada 2010]

[15] **MySQL AB**, [Enlace: <https://www.mysql.com/>], [Consultada 2010]

[16] **The PHP Group**, <http://www.php.net/>, [Consultada : 2010]

[17] **W3**, <http://www.w3.org/TR/REC-xml/>, [Consultada: 2010]

[18] **Cao, L., Yu, P. S., Zhang, C., & Zhao, Y.** (2010). Domain Driven Data Mining, 145–169. doi:10.1007/978-1-4419-5737-5

- [19] **Breslin, J. G., O’Sullivan, D., Passant, A., & Vasiliu, L. (2010)**. Semantic Web computing in industry. *Computers in Industry*, 61(8), 729–741.
doi:10.1016/j.compind.2010.05.002
- [20] **W3**, http://es.wikipedia.org/wiki/Interfaz_de_entrada_com%C3%BAn , [Consultada 2011]
- [21] **David Lowe**, http://en.wikipedia.org/wiki/Scale-invariant_feature_transform .
[Consultada: 2011]
- [22] http://en.wikipedia.org/wiki/Optical_character_recognition , “Definición y historia del reconocimiento de caracteres”[Consultada 2011]
- [23] **Lockwood, F.G., Williams, A.I... & Roberts, D.W. (1988)**. Improving Teaching at a Distance within the University of the South Pacific. *International Journal of Educational Development*, 8(3), 265-267.
- [24] **Chambers, Ellis. (1992)**. Work-load and the quality of student learning. *Studies in Higher Education*, 17(2), 141-153p
- [25] **Mckay, R. (1978)**. Effectiveness of learning: the place of study, En: D. WARREN PIPER (Ed.) *The Efficiency and Effectiveness of Teaching in Higher Education*, pp. 86-94. London: University of London Institute of Education.
- [25] **Masahiro Hara**, <http://www.qrcode.com/en/index.html> , [Consultada: 2011]
- [26] **The Open Source Initiative**, <http://opensource.org/> , [Consultada: 2012]
- [27] **Free Software Foundation, Inc.**, <http://www.gnu.org/copyleft/gpl.html> , [Consultada: 2012]
- [28] **W3**, http://es.wikipedia.org/wiki/Servidor_web , [Consultada 2013]