



# DESARROLLO DE UN TUTOR INTELIGENTE PARA ASISTIR AL PROFESOR EN UN SISTEMA DE GESTIÓN DEL APRENDIZAJE

Mario Alejandro Campos Soberanis

Tesis elaborada para obtener el grado de Maestro en Ciencias de la  
Computación

Tesis dirigida por  
Víctor Hugo Menéndez Domínguez  
Alfredo Zapata González

Mérida, Yucatán  
Enero 2018



Declaro que esta tesis es mi propio trabajo, con excepción de las citas en las que he dado crédito a sus autores, asimismo afirmo que este trabajo no ha sido presentado para la obtención de algún título, grado académico o equivalente

Mario Alejandro Campos Soberanis

Agradezco el apoyo brindado por el Consejo Nacional de Ciencia y Tecnología (CONACYT) por haberme otorgado la beca No. 589429 durante el período de agosto de 2015 a julio de 2017 para la realización de mis estudios de maestría que concluye con esta tesis, como producto final de la Maestría en Ciencias de la Computación de la Universidad Autónoma de Yucatán.

## **Dedicatoria**

*Para mi familia, por nutrirme de los valores para alcanzar mi mayor tesoro: mi libertad de conciencia.*

*Para mis hermanos, hombres libres que me motivan siempre a transformarme en un mejor ser humano.*

*Para mi esposa, columna en la cual apoyo mi felicidad y logros.*

*A todos, infinitas gracias.*

## **Agradecimientos**

A mi asesor, el Dr. Víctor Hugo Menéndez Domínguez, por el invaluable apoyo ofrecido durante todo el proceso de elaboración de esta tesis y por brindarme la guía y correcciones necesarias para concluir el proyecto.

A mi coasesor, el Dr. Alfredo Zapata González por la enorme ayuda brindada para realizar la experimentación del proyecto, así como por sus valiosas aportaciones a lo largo del trabajo.

A los Dres. Francisco Romero Chicharro y José Ángel Olivas Varela de la Universidad de Castilla-La Mancha (UCLM), por el fraternal recibimiento y por su guía precisa para enriquecer el proyecto.

Al Departamento de Innovación e Investigación Educativa (DIIE) y la Facultad de Educación de la Universidad Autónoma de Yucatán (UADY) por la atención brindada al presente trabajo para realizar la validación del proyecto y por sus valiosas aportaciones para mejorar el prototipo.

A todo el personal académico, administrativo y manual de la Facultad de Matemáticas, por el incansable trabajo para brindar a quienes tenemos el gusto de pertenecer a la institución, el entorno propicio para desarrollarnos académica y personalmente.

## Resumen

El uso de sistemas inteligentes en los entornos educativos representa una oportunidad para mejorar los procesos de enseñanza y aprendizaje. Con este objetivo se han propuesto diferentes sistemas entre los que se encuentran tanto Sistemas de Gestión del Aprendizaje como Sistemas de Tutoría Inteligente (ITS).

Esta tesis se orienta al diseño de una arquitectura general para sistemas ITS que puedan ser incorporados en entornos de gestión del aprendizaje (LMS).

Para tal efecto se propone una metodología de trabajo y un marco arquitectónico para la creación de ITS integrados en entornos LMS y se describe el diseño, desarrollo y validación del sistema MITS (MOODLE Intelligent Tutoring System), el cual es un ITS conversacional diseñado para asistir a los profesores en el uso de las herramientas de la plataforma MOODLE.

Se describe la arquitectura e implementación del sistema, así como la validación de la usabilidad del mismo mediante la escala de usabilidad de sistemas (SUS) en un experimento aplicado con profesores de la Facultad de Educación y el Departamento de Innovación e Investigación Educativa de la Universidad Autónoma de Yucatán.

Los resultados del presente trabajo indican que los profesores encontraron útil y fácil de usar el sistema y opinan que puede resultar una herramienta efectiva para asistir a los diferentes usuarios dentro del sistema MOODLE.

**Palabras Clave:** Sistemas de Tutoría Inteligente, Sistemas de Gestión del Aprendizaje, MOODLE.

## Tabla de contenidos

viii

Dedicatoria.....	v
Agradecimientos .....	vi
Resumen.....	vii
Lista de tablas .....	x
Lista de figuras.....	xi
Capítulo 1. Introducción .....	1
1.1 Antecedentes y planteamiento del problema. ....	1
1.2 Pregunta de investigación, hipótesis y objetivos. ....	7
1.3 Método de trabajo. ....	9
1.4 Marco de trabajo. ....	11
1.5 Organización del documento. ....	11
Capítulo 2. Marco teórico .....	14
2.1. E-Learning .....	14
2.2. Tutores inteligentes.....	18
2.2.1. Antecedentes .....	18
2.2.2. Concepto de sistema de tutoría inteligente (ITS).....	21
2.2.3. Arquitectura de los ITS.....	23
2.2.4 Ventajas y desventajas de los ITS.....	27
2.2.5. Implementaciones de los ITS.....	28
2.3. Sistemas de gestión del aprendizaje.....	30
2.3.1. Sistemas de gestión del aprendizaje y competencias.....	38
2.3.2 Diseño instruccional en entornos virtuales de aprendizaje.....	46
2.4. Gestión del conocimiento. ....	49
2.4.1 Definiciones y taxonomía. ....	49
2.5. Minería de datos.....	52
2.6. Minería de datos en educación.....	58
2.7. Agentes conversacionales.....	62
2.8. Trabajos relacionados .....	64
Capítulo 3. Metodología y propuesta.....	70
3.1. Metodología de trabajo. ....	70
3.2. Propuesta: MOODLE Intelligent Tutoring System (MITS) .....	75
3.3. Arquitectura del sistema .....	91
3.4. Implementación técnica de la propuesta .....	96
Capítulo 4. Experimentación y resultados. ....	103
4.1 Pruebas automatizadas.....	103
4.2 Diseño del experimento. ....	105
4.3 Resultados.....	109
Capítulo 5. Conclusiones y trabajo futuro. ....	116
5.1Análisis del cumplimiento de los objetivos. ....	116
5.2 Conclusiones.....	118
5.3 Recomendaciones y trabajo futuro.....	120
5.4 Divulgación de resultados.....	123
Lista de referencias .....	125



## Lista de tablas

x

Tabla 1. Entidades del sistema MITS .....	80
Tabla 2. Intenciones del sistema MITS.....	81
Tabla 3. Secuencia de ejecución para responder preguntas del sistema MITS .....	100
Tabla 4. Secuencia de ejecución para emitir recomendaciones del sistema MITS .....	101
Tabla 5. Perfiles predefinidos de usuario del sistema MITS .....	104
Tabla 6. Resultados de pruebas automatizadas.....	105
Tabla 7. Escala de usabilidad del sistema SUS.....	108
Tabla 8. Interacciones no incluidas en el dominio de conocimiento del sistema MITS.....	110
Tabla 9. Resultados de la prueba SUS del sistema MITS.....	111

Figura 1. Proceso de evolución de los ITS. ....	21
Figura 2. Arquitectura general de los ITS.....	26
Figura 3. Proceso de transformación de datos en información.....	49
Figura 4. Proceso de transformación de datos en información.....	51
Figura 5. Taxonomía de los modelos utilizados en minería de datos. ....	54
Figura 6. Interacción de alumnos, académicos e investigadores con los sistemas educativos y el proceso de minería de datos (tomado de Romero y Ventura, 2007).....	59
Figura 7. Esquema de extracción de conocimiento e interacción del ITS con el profesor. ....	70
Figura 8. Esquema de comunicación e interacción entre el ITS y el profesor.....	72
Figura 9. Esquema de interacción del ITS con el profesor. ....	74
Figura 10. Interfaz del sistema MITS. ....	76
Figura 11. Interfaz del sistema MITS con elementos multimedia incrustados.....	77
Figura 12. Ejemplo de XML de intenciones.....	79
Figura 13. Ejemplo de XML de entidades.....	79
Figura 14. Ontologías de entidades del sistema MITS. ....	83
Figura 15. Construcción del contexto del usuario. ....	84
Figura 16. Construcción del contexto del usuario con información implícita.....	85
Figura 17. Pseudocódigo del algoritmo de extracción de contextos.....	85
Figura 18. Diagrama de flujo del algoritmo de extracción de contextos.....	87
Figura 19. Componentes del sistema MITS.....	92
Figura 20. Diagrama de clases del sistema MITS.....	95
Figura 21. Implementación del sistema MITS.....	97
Figura 22. Tabla de contextos del sistema MITS. ....	98
Figura 23. Tabla de interacciones del sistema MITS.....	99
Figura 24. Gráfica de interacciones no incluidas en el dominio de conocimiento del sistema MITS.....	111
Figura 25. Gráfica de resultados de la escala SUS del sistema MITS.....	113
Figura 26. Gráfica de promedio de resultados por pregunta de la escala SUS del sistema MITS.....	113

## **Capítulo 1. Introducción**

Este capítulo tiene como propósito ofrecer una visión general del trabajo realizado en esta tesis. Se inicia describiendo el entorno de la investigación, resaltando los problemas relacionados con el desarrollo e implementación de tutores inteligentes que han dado origen a la hipótesis y los objetivos de esta investigación. A continuación, se introduce el método y el marco de trabajo empleado dentro de la investigación. Finalmente se presenta la estructura organizativa del documento.

### **1.1 Antecedentes y planteamiento del problema.**

En la actualidad existen necesidades de formación crecientes que no pueden ser satisfechas con los métodos tradicionales de enseñanza, debido a lo cual los modelos de educación en línea están cobrando cada vez mayor relevancia.

El desarrollo del e-learning ha mostrado grandes avances gracias al esfuerzo de diferentes instituciones educativas y comerciales a nivel internacional que han encontrado beneficios mediante el uso de la enseñanza en línea y han comenzado a aplicarla para satisfacer diferentes necesidades.

La educación en línea es un concepto que ha ido evolucionando a través del tiempo desde los primeros acercamientos a la educación a distancia que tuvieron lugar a mediados del siglo XIX, cuando Isaac Pitman, propuso un sistema para entrega de trabajos y educación por correspondencia con estudiantes Británicos (Lease y Brown, 2009).

El sistema de correspondencia se difundió rápidamente y algunos programas de educación superior como el de la Universidad de Illinois Wesleyan ofrecieron grados para estudiantes en ausencia, siendo probablemente el de la Universidad de Chicago el primer programa de educación superior a distancia (Kerka, 1996; Mcisaac y Gunawardena, 1996).

Sin embargo, la educación a distancia generaba inseguridad acerca de la calidad de sus programas y en las décadas de los años 1920 y 1930 las escuelas por correspondencia en los Estados Unidos tuvieron una reputación tan mala que cambiaron el nombre de sus programas de educación a distancia a “Estudio guiado independiente” (Bates 2000).

Las formas de la educación a distancia fueron adaptándose con el desarrollo de los nuevos métodos y tecnologías surgidas en los siglos XIX y XX (Valentine, 2002). Utilizando las tecnologías emergentes la educación a distancia evolucionó incorporando cintas de radio, conferencias vía telefónica, televisión y videoconferencia para sustituir el sistema por correspondencia mejorando las capacidades interactivas de la educación a distancia (Lease y Brown, 2009).

Con el desarrollo del transistor y la computadora surgió un auge en su uso como un método de para la instrucción a distancia. La mejora exponencial de la computadora el desarrollo de internet y de programas de software como Blackboard, WebCT y Lotus Notes desembocaron en el crecimiento exponencial de la instrucción basada en web (Bates, 2000).

De manera creciente internet se ha convertido en el medio preferido para la educación a distancia. (National Center for Education Statistics, 2002).

El concepto de educación a distancia ha derivado a especializaciones del mismo siendo el e-learning una de las más relevantes hoy en día.

El desarrollo del e-learning ha llevado a la propuesta de varias definiciones algunas enfatizando los aspectos del uso de internet y del uso de la tecnología respectivamente. Al respecto Rosenberg (2001) define e-learning como “el uso de las tecnologías de internet para entregar un amplio conjunto de soluciones que mejoran el conocimiento y desempeño” mientras que el término es definido por Rosset y Sheldon (2001) como “una amplia combinación de procesos, contenidos e infraestructura para usar las computadoras y redes para escalar o mejorar una o más partes significativas de una cadena de valor de aprendizaje, incluyendo administración entrega”.

A partir de la definición de Rosset y Sheldon, es claro que el e-learning involucra la gestión del conocimiento y el proceso por medio del cual éste es puesto al alcance del educando, por lo cual los Sistemas de Gestión del Conocimiento (Learning Management Systems LMS), se han convertido en una herramienta fundamental para el e-learning. Con el desarrollo del nuevo paradigma de educación en línea surgen nuevos retos para adaptar el proceso de enseñanza y aprendizaje en éste nuevo entorno, siendo una de sus partes fundamentales el diseño instruccional.

Según Gordillo y Chávez (2007) el diseño instruccional es el proceso en donde se analizan, organizan y presentan objetivos, información, actividades, métodos, medios y el

proceso de evaluación, que al conjugarse entre sí conforman el contenido de un curso con miras a generar experiencias satisfactorias de aprendizaje.

Existen algunas iniciativas para generar herramientas para el diseño instruccional en la educación superior, como el Generador de Diseño Instruccional propuesto por Francesa, Espinoza y Chacón (2012), mismo que se basa en un LMS implementado en el Tecnológico de Costa Rica.

La Universidad Autónoma de Yucatán (UADY) ha implementado políticas para el uso de las TIC en sus procesos académicos y administrativos, las cuales se contemplan en su Plan de Desarrollo Institucional (PDI) 2010-2020, PDI 2014-2022 y en su Modelo Educativo para la Formación Integral (MEFI).

En el PDI del 2010-2020 (2010) la universidad plantea lo siguiente: “... la formulación de nuevos programas de licenciatura y posgrado utilizando modalidades no presenciales y semipresenciales, privilegiando la equidad y el uso de tecnologías de información y comunicación. La nueva oferta debe considerar su contribución a la operación de los programas presenciales, así como a la formación de los estudiantes inscritos en ellos” (p.76).

Por su parte, el MEFI (2012) señala: “... la planeación deliberada y sistemática de nuevas propuestas para la solución de situaciones problemáticas y para la mejora continua de la práctica educativa que implica un cambio en el contexto y la práctica educativa misma, mediante la incorporación de recursos y medios educativos vanguardistas...” (p.39); respecto al profesor universitario: “14. Utilizar las tecnologías

de información y comunicación como recurso didáctico en su ejercicio docente de manera pertinente.” (p.49).

En respuesta a lo anterior, en 2011, la Universidad integra al sistema de educación en línea (modalidad mixta), la educación virtual (modalidad no presencial) lo que da origen a un entorno virtual de aprendizaje denominado UADY Virtual (Plan Desarrollo UADY Virtual, 2013, p.31).

El LMS denominado UADY Virtual, en conjunto con el MEFI, plantean el siguiente objetivo: “Impulsar la innovación académica de la institución en todos los niveles, a través de las diferentes modalidades educativas contempladas en el MEFI, especialmente con el apoyo de entornos virtuales de aprendizaje y la puesta en marcha de las modalidades mixta y no presencial (modalidades no convencionales).” (Plan Desarrollo UADY Virtual, 2013, p.31).

Recientemente la UADY anunció la creación de cuarenta y cinco nuevas licenciaturas en línea, lo cual implica el involucramiento de toda la comunidad universitaria en un nuevo paradigma educativo que demanda nuevas competencias por parte de docentes y estudiantes (<http://sipse.com/milenio/uady-licenciaturas-linea-estudios-educacion-internet-carrera-193058.html>).

Debido a lo anterior es claro que las necesidades de formación docente deben hacer énfasis en el uso de las herramientas utilizadas para el e-learning resultando de primera importancia aquellas que se encuentran en las plataformas en las cuales se están implementando éstos nuevos modelos de formación académica.

Sin embargo, la implementación de estrategias de formación docente no es una tarea sencilla puesto que la Universidad presenta un ambiente heterogéneo en cuanto a las competencias en el uso de las tecnologías para la educación en línea y resulta complicado adaptar el aprendizaje a éste conjunto de profesores con diferente formación en el ámbito del e-learning.

Una estrategia que pudiese resultar de ayuda para personalizar la formación de los docentes es usar el aprendizaje adaptativo el cual es definido por Dāboliņš (Dāboliņš, 2013) como el proceso de generar una experiencia de aprendizaje única para cada educando basada en su personalidad, intereses y desempeño para alcanzar metas como el mejoramiento académico del estudiante, la satisfacción del usuario, efectividad en el proceso de aprendizaje, entre otras.

El mismo Dāboliņš (Dāboliņš, 2013) menciona que los Sistemas de Tutoría Inteligente (ITS) son sistemas y tecnologías que utilizan tecnologías de aprendizaje adaptativo para personalizar el proceso de aprendizaje de acuerdo con las necesidades del individuo, tomando en cuenta factores del conocimiento del tema por parte del estudiante, sus emociones y estilo de aprendizaje.

El uso de estos sistemas permite añadir flexibilidad a los materiales de aprendizaje, modificando el enfoque didáctico de acuerdo con las competencias y particularidades del estudiante y la tarea de aprendizaje.

Debido a la capacidad de los ITS para adaptarse a las necesidades del usuario, resultan de gran utilidad dentro de entornos con necesidades heterogéneas de formación y en el caso particular de la UADY ofrecen un área de oportunidad para apoyar en la

formación de los docentes para interactuar con las plataformas de Gestión del Conocimiento en las cuales se encuentran trabajando.

El presente trabajo de investigación tiene por objeto el desarrollo, implementación y evaluación de un ITS para asistir al profesor en el desarrollo de una actividad específica en el LMS, con el fin de aportar a la formación docente en lo concerniente al uso de las herramientas proporcionadas por el LMS.

Los resultados de esta investigación permitirán determinar si los Sistemas de Tutoría Inteligente pueden ser de utilidad en la formación docente, en particular en el proceso de profundizar en el manejo de los LMS. Lo anterior puede resultar de gran utilidad para las instituciones con necesidades formativas de competencias en e-learning para su personal docente, mediante la extensión de los resultados de esta investigación a nuevas herramientas que complementen la formación docente e inclusive puede servir para perfilar un marco de trabajo y un entorno de desarrollo genérico para la creación de diversos ITS con objetivos específicos en entornos LMS.

## **1.2 Pregunta de investigación, hipótesis y objetivos.**

A continuación, se presenta la hipótesis, el objetivo general y los objetivos específicos del proyecto. A partir de la pregunta de investigación:

### **Pregunta de investigación**

¿Es posible facilitar el uso de las herramientas de un sistema de gestión del aprendizaje

de un profesor universitario a través del uso de tutores inteligentes?

Se pretende analizar si el uso de tutores inteligentes puede mejorar el desempeño de un profesor bajo el supuesto de que un mejor conocimiento de las herramientas ofrecidas por los LMS a través de un tutor inteligente permitirá al profesor optimizar sus tiempos y por lo tanto mejorar su desempeño. Se plantea la siguiente Hipótesis:

**Hipótesis inicial.**

Los tutores inteligentes para el uso de herramientas implementados en un sistema de gestión del aprendizaje facilitan el uso de dichas herramientas.

A partir de la hipótesis establecida y una vez estudiados los aspectos relacionados con la problemática presentada, se indica el objetivo general de este trabajo:

**Objetivo general**

Desarrollar y evaluar un tutor inteligente para asistir al profesor en el uso de herramientas dentro del entorno de un sistema de gestión del aprendizaje.

Este objetivo principal se detalla a través de los siguientes objetivos específicos:

1. Analizar sistemas de gestión del aprendizaje disponibles y seleccionar uno que ofrezca herramientas robustas para la implementación de un curso en línea y permita la integración y desarrollo de módulos externos de manera que sea posible el desarrollo de un tutor inteligente integrado al mismo.
2. Determinar las mejores formas de asistir al profesor mediante el uso de un tutor inteligente, para desarrollar su conocimiento de las herramientas disponibles para la impartición de un curso en el sistema de gestión del aprendizaje elegido.
3. Desarrollar un tutor inteligente integrado al sistema de gestión del aprendizaje seleccionado que implemente las formas adecuadas de asistir a los profesores en el uso de las herramientas dentro de un curso.
4. Realizar los experimentos necesarios para verificar la efectividad de la propuesta, basados en cantidades suficientes de datos obtenidos a partir del uso de la propuesta por parte de profesores reales.

### **1.3 Método de trabajo.**

El método de trabajo utilizado para el desarrollo de esta tesis sigue el enfoque de la Investigación-Acción en Sistemas de Información, el cual se ha convertido en uno de los principales métodos de investigación cuantitativa en el campo de los sistemas de información y en la ingeniería del software (Ruiz et al., 2002).

A continuación, se describen de forma general las principales fases consideradas para el desarrollo del presente trabajo:

1. **Definición de la problemática.** Se refiere a la determinación de forma eficiente de la problemática asociada a los Sistemas de Tutoría Inteligente (en inglés Intelligent Tutoring Systems, ITS). Esto ha originado la necesidad de estudiar las diferentes metodologías que involucran técnicas de desarrollo e implementación de los ITS.
2. **Análisis del estado del arte.** Donde se ha realizado una revisión sistemática (Biolchini et al, 2005) de las temáticas afines a los LMS, escenarios y actividades de aprendizaje, técnicas de extracción de conocimiento, técnicas de sistemas de recomendación, entre otros. La revisión sistemática contiene una secuencia estricta y bien definida de pasos metodológicos, de acuerdo con un protocolo definido a priori (Kitchenham, 2007).
3. **Propuesta de un marco metodológico para la construcción de un ITS.** La propuesta se ha dividido en las siguientes actividades:
  - a. Definición de un marco metodológico para recabar la información necesaria para el desarrollo de ITS.
  - b. Propuesta de una arquitectura para el desarrollo de un ITS.
  - c. Propuesta de un sistema de tutoría inteligente para asistir al profesor en el uso de las herramientas del sistema MOODLE.
4. **Implementación de un prototipo.** Esta fase comprende las siguientes actividades:

- a. Desarrollar una aplicación software de tutoría inteligente.
  - b. Implementar la aplicación software dentro de un sistema de gestión del aprendizaje de una arquitectura abierta y distribuida.
5. **Experimentación.** Se ha realizado un estudio empírico del método y los sistemas desarrollados para demostrar su validez. Para ello, se han llevado a cabo experimentos con datos reales en cantidades suficientes provenientes de profesores y alumnos de la Universidad Autónoma de Yucatán, México.

#### **1.4 Marco de trabajo.**

El presente trabajo de tesis es uno de los diversos proyectos que se gestan en el grupo académico de Ingeniería de Software e Informática Educativa, el cual tiene como objetivo la investigación e innovación de los procesos de desarrollo, operación y mantenimiento de Software, así como en las aplicaciones educativas para asistir los procesos de enseñanza y de aprendizaje (FMAT-UADY, 2015). Sus líneas de investigación se enfocan en:

- 1 Ingeniería de Software: Esta línea tiene como propósito el estudio y aplicación de enfoques sistemáticos, disciplinados y cuantificables para el desarrollo, operación y mantenimiento de Software.
- 2 Informática Educativa: Esta línea tiene como propósito el estudio del uso, efectos y consecuencias de las tecnologías de la información en el proceso educativo.

#### **1.5 Organización del documento.**

El presente trabajo de tesis se estructura en cinco capítulos (ver Figura 1.1). Estos capítulos describen distintos aspectos del trabajo realizado: revisión documental, trabajos

relacionados, metodologías, implementaciones y validaciones. A continuación, se detallan los contenidos:

**Capítulo 1: Introducción.** Este capítulo presenta una visión general del trabajo. Se inicia indicando el marco de la investigación, la hipótesis y los objetivos perseguidos en la realización de esta investigación. Se establece el método de trabajo y los proyectos relacionados. Por último, se exponen las aportaciones de esta investigación.

**Capítulo 2: Marco teórico.** En este capítulo se presenta una revisión documental relativa a las temáticas más relacionadas y que resultan primordiales para la investigación desarrollada en este trabajo: conceptos, definiciones y metodologías para el desarrollo e implementación de tutores inteligentes, definiciones y funcionamiento de los sistemas de gestión del aprendizaje, conceptos y procesos involucrados en la gestión y descubrimiento de conocimiento y técnicas de minería de datos.

**Capítulo 3: Metodología y propuesta.** En este capítulo se describe la metodología de trabajo empleada para recolectar la información necesaria, así como la propuesta del sistema y la implementación técnica de un prototipo.

**Capítulo 4: Experimentación y resultados.** Este capítulo describe los experimentos realizados para la validación del modelo propuesto. Para ello, se establecen medidas que comprueban la efectividad y el rendimiento del sistema propuesto. También se consideran cuestionarios para evaluar la usabilidad del sistema.

**Capítulo 5: Conclusiones y trabajo futuro.** En este capítulo se describen las principales aportaciones de esta tesis, el cumplimiento de los objetivos planteados y las principales líneas de trabajo futuro que se abren a partir de esta investigación.



## Capítulo 2. Marco teórico

El objetivo del presente capítulo es introducir los antecedentes relacionados con el tema de investigación de esta tesis. Primero se describen los conceptos más importantes relacionados al e-learning. Posteriormente se presentan conceptos, definiciones y metodologías para la implementación y desarrollo de Tutores Inteligentes eje central del trabajo de esta tesis. En tercer lugar, se aborda el tema de sistemas de gestión del aprendizaje. Posteriormente, se abordan definiciones y técnicas de gestión del conocimiento. Finalmente, se abordan temáticas de minería de datos web con énfasis en su aplicación en entornos educativos.

### 2.1. E-Learning

Los avances en las tecnologías de la información junto con los cambios en la sociedad han dado lugar a la creación de nuevos paradigmas para la educación como el e-learning, en el que la web juega un papel importante (Khan, 1997). El concepto de e-learning está asociado con actividades de aprendizaje que impliquen el uso de redes y computadores en forma simultánea (Tsai y Machado, 2002). El término e-learning puede definirse de una forma simple como el empleo de internet para facilitar el aprendizaje. Esto implica utilizar las herramientas disponibles en internet y más específicamente en la web, para adaptar el ritmo de aprendizaje al estudiante sin considerar los límites de espacio y tiempo (Brogan, 1999). Existen otras modalidades del e-learning tales como:

**E-training.** Es aquella opción formativa, compuesta de cursos cortos hasta un nivel de diploma o certificado, tiene como objetivo la actualización de conocimientos o

habilidades en un área sumamente específica, ya sea de un campo disciplinar o bien relacionada con ciertas destrezas y habilidades laborales (Sánchez et al., 2006).

**Web-based education.** Es la modalidad más extrema del e-learning, debido a que utiliza el streaming de vídeos y las funcionalidades más avanzadas disponibles en el software educativo y donde no hay contacto real de cara a cara entre el profesor y el alumno (Lynch, 2002).

**Mobile education.** Es una extensión del e-learning que se realiza a través de dispositivos móviles de comunicación, por ejemplo: teléfonos inteligentes, asistentes personales digitales, tabletas, entre otros (Traxler, 2005).

Conviene considerar en e-learning, dos aspectos principales:

**Educativo.** Está relacionado con la necesidad de contar con un diseño instruccional sólido y acorde con las consideraciones pedagógicas, las características del medio y el comportamiento de los usuarios.

**Tecnológico.** Se trata de las herramientas que se ofrecen para la presentación de los contenidos; la comunicación entre usuarios o la evaluación, entre otras.

Dado el punto de vista tecnológico, el e-learning ha evolucionado, de simples páginas HTML (HiperText Markup Language) hasta complejos sistemas de gestión del aprendizaje. Se pueden identificar cinco tipos de tecnologías e-learning que pueden ser utilizadas solas o combinadas (Kapp, 2003):

**Lenguajes de programación.** Permiten tener un completo control y flexibilidad en los mecanismos de interacción del aprendizaje en línea. Sin embargo, una de sus desventajas es que el mantenimiento de tales sistemas es una tarea que requiere contar con personal especializado.

**Paquetes de autoría.** Emplean elementos, muchas veces visuales, que facilitan la construcción de aplicaciones e-learning sin necesidad de contar con conocimientos de programación. Su principal desventaja es que ofrecen una funcionalidad limitada y generalmente no incluyen los elementos de interacción propios de un sistema e-learning. Además, suelen estar diseñados para un lenguaje de programación específico.

**Sistemas de Gestión del Contenido (CMS, Content Management System).** Permiten tener control y orden sobre los recursos educativos debido a que todos los elementos de información están catalogados. Esto permite una búsqueda y exploración del contenido de forma sencilla.

**Sistemas de Gestión del Aprendizaje (LMS, Learning Management System).** Están diseñados para asistir al proceso de enseñanza-aprendizaje en un ambiente en línea mediante un conjunto de herramientas. Pueden tener una orientación hacia el alumno o el profesor.

**Sistemas de Gestión de Contenidos de Aprendizaje (LCMS, Learning Content Management System).** Son sistemas independientes o integrados con un LMS, que gestionan y administran los contenidos de aprendizaje. Una vez que los contenidos están en este sistema, pueden ser combinados, asignados a distintos cursos o descargados, entre muchas otras opciones.

De todas las tecnologías presentadas, las referidas a los LMS son las más empleadas actualmente por numerosas instituciones para organizar y distribuir cursos en línea (Kapp, 2003). Generalmente los LMS son fáciles de usar, se consideran flexibles en términos pedagógicos y eficientes con relación a los costos. Además de permitir organizar los contenidos educativos a la vez ofrece una serie de funcionalidades de comunicación (Downes, 2005). Algunos ejemplos de LMS son:

**MOODLE** (Modular Object Oriented Distance Learning Environment). Es un sistema de gestión de cursos de aprendizaje en línea de distribución libre. Su diseño está basado del enfoque pedagógico constructivista social. Posee una arquitectura modular, lo que permite incorporar una gran diversidad de componentes y funcionalidades. Implementa una interfaz de navegador de tecnología sencilla, ligera, eficiente y compatible (Dougiamas y Taylor, 2003).

**Dokeos**. Es un sistema de aprendizaje en línea basado en software libre. Posee una interfaz intuitiva que facilita la creación y organización de contenidos interactivos y ejercicios, así como también, incorpora herramientas de seguimiento e informes sobre el desempeño de los alumnos en el curso. Su arquitectura está basada en módulos lo que facilita la incorporación de nuevas funciones (De Praetere, 2010).

**ILIAS** (Integrated Learning, Information and cooperAtion System). Es un sistema de gestión del aprendizaje de código abierto. Posee un ambiente de aprendizaje flexible, el cual permite la reutilización y el intercambio de materiales de instrucción entre profesores y alumnos. Destaca su sistema de control de acceso basado en roles y la facilidad para la creación de cuestionarios (University of cologne, 2004).

**ATutor.** Es un sistema de código abierto que se utiliza para la impartición de cursos en línea. Su instalación es rápida y pueden desarrollarse temas para cambiar la apariencia de la plataforma, así como instalar nuevos módulos para ampliar las funcionalidades de la misma. Los profesores pueden crear, configurar y empaquetar cursos para su redistribución en la red y los estudiantes aprenden en un entorno adaptativo y social (Toronto University, 2002).

## **2.2. Tutores inteligentes**

En el presente apartado se examinan los antecedentes, conceptos, arquitectura ventajas, desventajas y uso de los Sistemas de Tutoría Inteligente (Intelligent Tutoring Systems, ITS).

### **2.2.1. Antecedentes**

Junto con el avance de la tecnología computacional, la forma de educar ha evolucionado y a lo largo de los años se han realizado diversas propuestas para utilizar dicha tecnología en el ámbito de la educación.

Con el desarrollo de la tecnología computacional y multimedia surgió el concepto de entrenamiento basado en cómputo (en inglés Computer Based Training, CBT) y el de instrucción asistida por computadora (en inglés Computer Assisted Instruction, CAI) los cuales utilizaban las capacidades interactivas multimedia y la portabilidad de los CD-ROMs para proveer entrenamiento en horarios flexibles usando materiales desarrollados en la computadora, permitiendo reducir costos de entrenamiento. Sin embargo, a pesar de

estos avances la falta de un tutor hacía la experiencia más lenta y menos atractiva para los estudiantes (Keles y Keles, 2011).

El desarrollo del internet trajo consigo grandes oportunidades para muchas áreas entre ella la educación, surgiendo la educación basada en web (en inglés Web Based Education, WBE) la cual presentaba varias ventajas como la conveniencia de tomar el curso sin salir del trabajo o la casa y costos reducidos (Berz, Erdeyli y Hoefkens, 1999).

El desarrollo del internet y el enriquecimiento de las aplicaciones computacionales permitieron revolucionar la educación basada en web creando ambientes que mejoraban la calidad de la experiencia educativa ocasionando el desarrollo de la siguiente generación de sistemas de aprendizaje en línea (Keles y Keles, 2011).

A pesar de que los sistemas de aprendizaje en línea eran adoptados por varias organizaciones, presentaban algunos obstáculos como la dificultad de evaluar su desempeño, el cambio continuo del comportamiento de los estudiantes, dificultad en diseñar un sistema de aprendizaje en línea capaz de ajustarse a las necesidades de los usuarios, costos y tiempos de desarrollo elevados (Keles y Keles, 2011).

Como un esfuerzo para sobrellevar estas dificultades se desarrollaron programas inteligentes de instrucción asistida por computadora (en inglés Intelligent Computer Assisted Instruction, ICAI), los cuales incorporaban técnicas de inteligencia artificial (en inglés Artificial Intelligence, AI), como representación del conocimiento y procesamiento de lenguaje natural para adaptarse mejor a las necesidades e intereses de los estudiantes (Duchastel e Imbeau, 1988).

El concepto de ICAI fue introducido por Jaime Carbonell en 1970 cuando diseñó el programa llamado SCHOLAR. Este programa fue considerado también el primer ITS, debido a que separaba las capacidades del tutorial del conocimiento del dominio del sistema (Wegner, 1987).

La meta de los investigadores del área de ICAI era proveer a cada estudiante con un tutor por computadora que tenga todas las cualidades de un profesor experto, es decir un gran alcance y dominio de la materia, conocimiento excelente de técnicas de enseñanza y la habilidad de inspirar y motivar a los estudiantes para aprender (Kearsley, 1987).

Muchos investigadores en el área optaron por adoptar el término ITS en lugar de ICAI argumentando que la evolución del concepto y el cambio en la metodología de investigación iba más allá de la introducción de una I para el concepto de inteligencia (Wegner, 1987), esto fue confirmado cuando en 1988 se llevó a cabo la conferencia internacional en Sistemas de Tutoría Inteligente en Montreal, Canadá. Sin embargo, algunos investigadores mostraron reticencia al uso de la palabra inteligente por lo que optaron por etiquetas como sistema de tutoría basado en conocimiento (en inglés Knowledge Based Tutor System, KBTS), sistema de tutoría adaptativo (en inglés Adaptive Tutoring System, ADS) o sistema de comunicación de conocimiento (en inglés Knowledge Communication Systems, KCS); sin embargo, la mayoría de los investigadores se mostraron razonablemente satisfechos con el uso del acrónimo ITS (Keles y Keles, 2011).

El hecho de que los investigadores en el área de ITS se distribuyan entre varios campos implica que hay diferencias en las metas de investigación, terminología, marcos teóricos y énfasis entre ellos (Hyacinth, 1990).

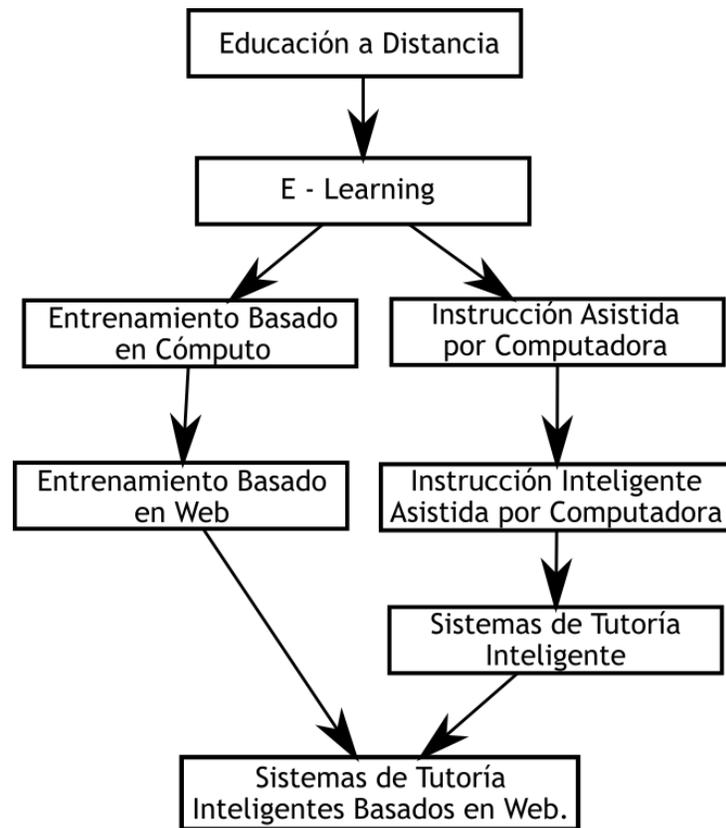


Figura 1. Proceso de evolución de los ITS.

### 2.2.2. Concepto de sistema de tutoría inteligente (ITS).

Según Stankov, Glavinic y Rosic (2011) los ITS son un tipo particular de sistemas de e-learning asíncrono basados en conocimiento, diseñados para dar soporte y mejorar el proceso de enseñanza y aprendizaje para cierto dominio de conocimiento respetando la individualidad del estudiante como en una tutoría tradicional “uno a uno”. Los ITS son basados en conocimiento porque poseen conocimiento acerca de su dominio de

conocimiento, conocimiento acerca de los principios de enseñanza y los métodos para utilizarlos y conocimiento acerca de los métodos y técnicas de modelado del perfil del estudiante durante el proceso de adquisición del conocimiento.

Keles y Keles (2011) afirman que los ITS son programas de computadora diseñados para incorporar técnicas de la comunidad de AI para proveer tutores que sepan lo que enseñan, a quién lo enseñan y cómo enseñarlo.

Según Urban-Lurain (2003) los ITS se caracterizan por la habilidad inteligente de adaptar sus estrategias de enseñanza al conocimiento y habilidades de sus alumnos individuales de manera que la instrucción sea tan efectiva y eficiente como sea posible.

Lee y Sapiyan (2006) definen ITS como un software avanzado de entrenamiento que imita a un tutor humano para proveer aprendizaje individualizado.

De forma más específica podemos definir ITS como un sistema que evalúa acciones del usuario y desarrolla un modelo del conocimiento, habilidades y dominio de la materia del estudiante, para proveer pistas, ofrecer explicaciones y demostraciones basadas en el modelo obtenido. Éste enfoque es defendido por Romero y Ventura (2007) así como por Keles, Ocaik y Gülcü (2009).

Como vemos a pesar de la variedad de definiciones existentes en la literatura todas tienen en común la capacidad de los ITS de generar un modelo del conocimiento, habilidades y perfil de usuario y de adaptar el aprendizaje de forma individual utilizando técnicas de aprendizaje apropiadas para cada usuario. Muchos autores señalan como componentes clave de los ITS la extracción del perfil del usuario, el conocimiento de técnicas de enseñanza y la adaptabilidad del proceso de enseñanza y aprendizaje. A

menudo se menciona que el conocimiento pedagógico y el conocimiento de la materia son componentes separados dentro del sistema y el ITS tiene la capacidad de usar ambos para personalizar el proceso de enseñanza y aprendizaje.

### **2.2.3. Arquitectura de los ITS.**

Existe variedad en las arquitecturas utilizadas para implementar ITS, de hecho no es común encontrar dos ITS con la misma arquitectura, lo cual es un resultado de la naturaleza experimental del trabajo en el área (Keles y Keles, 2011), por lo que no existe una clara división para una arquitectura general para los ITS (Yazdani, 1986), pues algunos son desarrollados con base en teorías psicológicas (Künzel y Hämmer, 2006) otros se desarrollan para dominios de aplicación específicos (Bergin y Fors, 2003) y algunos más son utilizados para realizar pruebas de concepto de científicos de la computación (Martens y Himespach, 2005). Esta heterogeneidad en la aplicación de los dominios de aplicación de los ITS aumenta las diferencias entre sus arquitecturas, sin embargo, hay un consenso general en la literatura que identifica al menos tres componentes básicos para los ITS (Bonnet, 1985):

- El módulo de conocimiento de la materia.
- El módulo de conocimiento del estudiante.
- El módulo de tutoría.

En la mayoría de las implementaciones modernas de los ITS se añade además un cuarto componente:

- El módulo de interfaz.

El módulo de conocimiento de la materia representa el conocimiento del dominio de estudio para el cual el ITS es implementado. Es la parte que contiene reglas, información, preguntas y soluciones relevantes a la materia que será presentada al estudiante. Durante los últimos años se han desarrollado varios formalismos en AI para la representación del conocimiento: reglas simbólicas, grafos conceptuales, lógica difusa, redes bayesianas, redes neuronales, razonamiento basado en casos, minería de datos, etc. La mayoría de estas técnicas han sido utilizadas para representar el conocimiento en los ITS (Bulut, Akoplat y Orhan, 2013).

El módulo de conocimiento del estudiante recolecta información acerca del estudiante y el grado de dominio que tiene sobre la materia de estudio, conceptos correctos y erróneos, así como información acerca de las preferencias y estilo de aprendizaje del estudiante (Lavandelis y Bicans, 2011). Técnicas comunes para extraer el modelo del estudiante incluyen redes bayesianas y lógica difusa (Pek y Poh, 2005) mientras que otras se apoyan en el análisis de actividades o “traza” del estudiante (Hafdi y Bensebaa, 2013) y en cuestionarios para determinar el estilo de aprendizaje (Bulut, Akoplat y Orhan, 2013).

El módulo pedagógico o de tutoría contiene información pedagógica acerca de la forma de enseñar. Este módulo contiene estrategias de enseñanza e instrucciones para implementar el proceso de enseñanza (Pek y Poh, 2005). Existen algunas implementaciones interesantes de este módulo desde la perspectiva de psicología de la educación que utilizan estrategias colaborativas y utilizan el sistema para generar actores pedagógicos capaces de producir aprendizaje por disturbio (Aimeur, Frasson y Dufort,

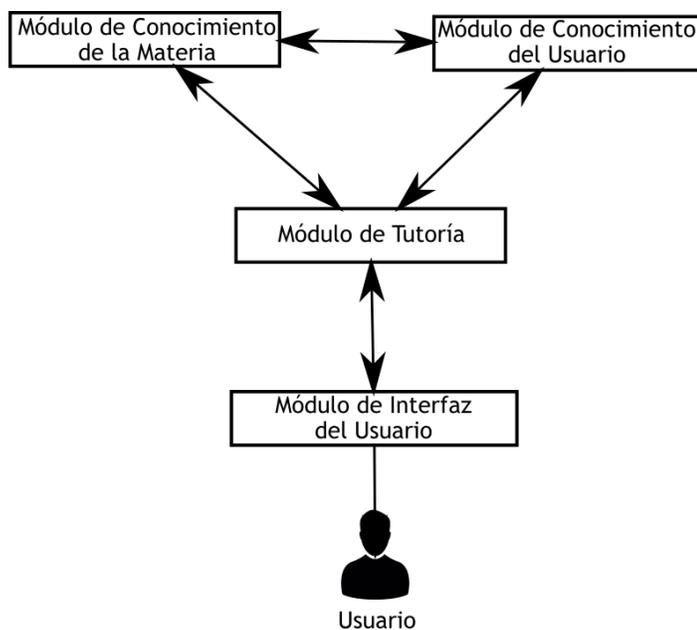
2000). Dabolins y Grundspenkins (2013) dividen los métodos de entrenamiento para el módulo de tutoría en tres grandes grupos:

**Monológicos:** Los métodos asumen que el concepto es más importante que el proceso de construcción y estos métodos son implementados como lecturas.

**Dialógicos:** Los métodos sugieren retroalimentación. El estudiante y el proceso de construcción son el foco principal de atención en estos métodos.

**Investigación:** El estudiante y el proceso son el foco de atención en un principio y el tutor juega el rol de un promotor y consultante. La retroalimentación generalmente es proporcionada junto con los resultados.

El módulo de interfaz se emplea a menudo como un canal de comunicación, bien sea entre los diferentes componentes del sistema o entre el sistema y el usuario.



*Figura 2. Arquitectura general de los ITS.*

Se han hecho algunas propuestas alternas de arquitectura que capturan los componentes básicos de los ITS pero son implementados desde perspectiva multiagente (Lavandelis y Bicans, 2011) o utilizando patrones de diseño y una arquitectura basada en componentes (Martens y Harrer, 2011).

Desde el punto de vista de las ciencias de la computación también se han hecho esfuerzos para formalizar la arquitectura de los ITS. Curilem, Barbosa y de Azevedo (2007) proponen una formalización para los ITS por medio de un autómata

$S = \{U, Y, X, x_0, \lambda, \eta\}$  en donde:

U: Conjunto de entradas formado por todas las posibles acciones que el estudiante puede realizar usando la interfaz.

Y: Conjunto de salida formado por los elementos que configuran la forma y contenido de la interfaz.

X: Conjunto de estados formado por todos los modelos de estudiante considerados por el sistema.

$x_0$  : Estado inicial representado por el modelo del estudiante oficial, obtenido usando un diagnóstico o estableciéndolo por default.

$\lambda : U \times X \rightarrow X$  : Función de transición de estados que actualiza el modelo actual del estudiante dependiendo de la entrada y el modelo actual del estudiante.

$\eta : U \times X \rightarrow Y$  : Función de salida que actualiza la configuración de la interfaz de acuerdo con la entrada y al modelo actual del estudiante.

#### **2.2.4 Ventajas y desventajas de los ITS.**

Los ITS proporcionan varias ventajas en comparación con enfoques tradicionales de educación y CBT entre las cuales encontramos las siguientes:

- Proporcionan materiales de aprendizaje acordes al estilo del aprendizaje, conocimiento, intereses y habilidades de cada alumno (Kozierkiewicz y Nguyen, 2011).
- Son capaces de adaptar el ambiente de aprendizaje a las preferencias de los usuarios (Kozierkiewicz y Nguyen, 2011).
- Otorgan a cada estudiante con un proceso familiar de enseñanza uno a uno (Alvi y Ali, 2010).
- Ofrecen pruebas adaptativas basadas en el nivel de conocimiento habilidad actual de cada usuario (Kozierkiewicz y Nguyen, 2011).
- Proveen accesibilidad a un gran número de usuarios potenciaes (Stankov, Rosic, Zitko y Grubisic, 2008).
- Propician el interés y motivación entre los estudiantes que los utilizan (Kozierkiewicz y Nguyen, 2011).
- No se cansan (Hafdi y Bensebaa, 2013).
- Dominio de los conceptos de la materia (Hafdi y Bensebaa, 2013).
- Bajos costos (Hafdi y Bensebaa, 2013).
- Independencia de tiempo, lugar y hardware (Dogan y Camurcu, 2010).

Las principales críticas desventajas de los ITS recaen en la dificultad y costo de desarrollo y mantenimiento ya que a menudo se necesita de un equipo interdisciplinario para desarrollar un ITS. Para el desarrollo de los diferentes módulos es necesario que psicólogos, científicos computacionales, expertos en el dominio de entrenamiento y diseñadores tengan que comunicarse para desarrollar el sistema (Martens y Harrer, 2011).

La complejidad de los ITS ha llevado a su implementación por medio de agentes, sin embargo la complejidad inherente en el sistema y las características de bajo acoplamiento organizacional hace que sólo una pequeña parte de las técnicas de análisis de requerimientos sean utilizadas de la ingeniería de software basada en agentes (en inglés Agent Oriented Software Engineering, AOSE) ocasionando que las metodologías de propósito general para el análisis de requerimiento sean difícilmente aplicables a los ITS (Lavandelis y Bicans, 2011).

Otro aspecto complejo en el desarrollo del ITS es el referente a la operación y mantenimiento del sistema debido a que se necesita un perfil experto debido a las técnicas de AI contenidas en el sistema (Bulut, Müzeyyen, Zuhtu y Ahmet, 2013).

### **2.2.5. Implementaciones de los ITS.**

Los ITS han sido utilizados en una amplia variedad de dominios de conocimiento entre los cuales se encuentran matemáticas, medicina, programación, música, física entre otros. A continuación, se presenta un listado de algunas de las propuestas de ITS existentes:

**SHERLOCK:** Utilizado para entrenar técnicos de la fuerza aérea para diagnosticar problemas en dispositivos complejos usados para dar servicio a los jets F-15. Presenta descripciones del dispositivo, diagramas de los circuitos y documentación del sistema. El tutor “crea” una falla en el dispositivo y le pide al alumno que diagnostique en que parte del circuito se encuentra la falla (Lajoie y Lesgold ,1989).

**Cardiac Tutor:** Entrena personal médico en técnicas avanzadas de soporte de vida cardíaco. Consiste en una simulación de la traza ECG del paciente, gases sanguíneos y signos vitales y el tutor provee pistas, avisos verbales y retroalimentación (Eliot y Woolf, 1994).

**Mathematics Tutor:** Enseña a un estudiante cómo resolver problemas de palabras que utilizan fracciones, decimales y porcentajes. El sistema clasifica los problemas matemáticos por operaciones, habilidades y complejidad. El sistema identifica el grado de desarrollo del estudiante y le propone problemas adecuados a su nivel (Beal, Beck y woolf, 1998).

**CyclePad:** Utilizado para la enseñanza del ciclo Rankine en termodinámica. Trabaja en modos de construcción, análisis y contradicción. Permite la conexión de elementos tomados del inventario del sistema usando un editor gráfico. Utiliza estrategias de enseñanza incrementales para el diseño y simulación de sistemas complejos (Wu, 1999).

**AutoTutor:** Utilizado para enseñar los principios fundamentales de hardware, sistemas operativos y el internet en un curso de introducción a la computación. Simula los patrones discursivos y estrategias pedagógicas de un tutor humano típico. Presenta

problemas de un guion de currículo y ofrece consejos y retroalimentación al estudiante (Graesser, 1999).

**PatternAdvisor:** Utilizado para la enseñanza de patrones de diseño de software a estudiantes. Asiste a los estudiantes recomendando las características necesarias para implementar patrones de diseño. Provee a los estudiantes con información teórica de los patrones elegidos (Berdun, Amandi y Campo, 2014.).

### **2.3. Sistemas de gestión del aprendizaje**

Existen varias maneras de alojar un LMS y que este funcione, las principales son: el uso de un sistema complejo elaborado dentro de una institución, el cual estaría diseñado para llevar acabo las funciones de gestionar cursos y el aprendizaje de los alumnos, el uso y adaptación de sistemas de LMS de código abierto y la contratación de licencias para sistemas de LMS de código cerrado o privativo. De acuerdo con Sánchez (2009), los LMS contienen una gran variedad de herramientas, tales como:

Distribución de contenido: es un espacio en donde el profesor comparte contenidos, por ejemplo: documentos, imágenes, videos, enlaces a sitios de interés, entre otros.

Comunicación y colaboración síncrona y asíncrona: en las herramientas asíncronas el estudiante controla el momento para el consumo del contenido, mientras que en las síncronas se requiere conexión a la plataforma en un momento determinado, y la utilización de los contenidos es un por un espacio de tiempo pre establecido. El objetivo de las anteriores permite que los participantes puedan comunicarse y trabajar

colaborativamente. Algunos ejemplos son: foros, salas de chat, mensajería interna, wikis, diarios, etc.

Seguimiento y evaluación: en este rubro destacan las pruebas generadas por el profesor y autoevaluaciones del alumno.

Administración y asignación de permisos: posibilitan la inscripción, y por consiguiente, el registro al curso.

Complementarias: contiene diversas ayudas tales como el portafolio, el bloc de notas, los sistemas de búsqueda de contenidos, entre otras.

Para complementar la información anterior, las características claves de los LMS son (Ahmad, et al., 2005):

- Administración de varios cursos o programas
- Llevar registro de los participantes
- Organizar y administrar el aprendizaje de los participantes
- Capacidad de proporcionar informes
- Programar y organizar los tiempos para el aprendizaje

Los LMS de acuerdo con su comercialización se clasifican en software propietario y de distribución libre. Los primeros, se caracterizan por que han sido desarrollados por una empresa particular. Su modelo de negocio se basa a través de la licencia de uso que se paga bajo un esquema mensual o anual (el costo depende del número de usuarios). Algunas de las ventajas son:

Suelen ser más estables y con funcionalidades que pueden adaptarse de acuerdo con las necesidades y el presupuesto.

Incluyen en muchos casos el alojamiento (servidores y ancho de banda) desligando al cliente de lidiar con estos requerimientos.

La capacitación de los profesores corre a cargo de la empresa contratada.

Algunas de sus desventajas son:

Suelen ser muy costosas: hay empresas que no solo cobran un mantenimiento mensual por el alojamiento en sus servidores sino también un costo mínimo de implementación.

No se tiene acceso al código fuente ya que las empresas no suelen vender el software, sino su implementación y posterior administración y mantenimiento, con el fin de asegurarse un abono mensual y/o anual.

Algunos ejemplos de Sistemas de Gestión del Aprendizaje de software propietario son:

- **Blackboard** es la plataforma de código privado líder en el mercado. Este LMS se caracteriza por poseer tecnologías educativas orientadas a una interfaz social, que permiten la participación activa y comprometida de sus participantes. Asimismo, cuenta con analíticas y herramientas de comunicación y soporte. Este sistema tiene presencia en más de 2200 instituciones educativas en más de 60 países. En México, diversas instituciones privadas lo utilizan, tales como el Tecnológico de Monterrey, Universidad Iberoamericana, Universidad Anáhuac, entre otras (Blackboard, 2016).

- **Dokeos** (De Praetere, 2010) en sus inicios fue un LMS de código abierto, sin embargo, hace algunos años cambio su filosofía y se convirtió en un software cerrado. Este sistema posee una interfaz intuitiva que facilita la creación y organización de contenidos interactivos y ejercicios, así como también incorpora herramientas de seguimiento e informes sobre el desempeño de los alumnos en el curso. Su arquitectura está basada en módulos lo que facilita la incorporación de nuevas funciones.
- **E-ducativa** es una empresa de tecnología informática orientada a la implementación de soluciones de e-learning desde hace más de 10 años. Este sistema está orientado al mercado Hispano. Actualmente cuenta con soluciones tanto para instituciones educativas como para empresas y gobiernos (e-ducativa, 2016).
- **Sumtotal** (Skillsoft Company, 2016) es un LMS con un alto grado de implantación en el ámbito empresarial. Incluye la posibilidad de integrarlo con la gestión de recursos humanos de la empresa. También cuenta con adaptación a dispositivos móviles.
- **eCollege** (Pearson education Inc, 2016) este LMS brinda a docentes y administradores las herramientas, el contenido y el apoyo que necesitan para crear, gestionar y medir experiencias de aprendizaje personalizadas y atractivas para sus estudiantes. Este sistema incluye herramientas para el apoyo del análisis de la gestión de programas educativos y ofrece seguimiento personalizado a los objetivos de aprendizaje.

Los LMS cuyo software es de distribución libre pueden instalarse tanto en computadoras personales como en servidores web. El soporte técnico suele estar soportado por una comunidad mundial de desarrolladores. Algunas de sus ventajas son:

Permite realizar modificaciones sobre el código del sistema, ya sea para personalizarlo o agregarle nuevos componentes.

- Son compatibles con la mayoría de los formatos estándares.
- Su actualización es constante.
- Ausencia de malware al momento de instalarlo.

En contraste, algunas de sus desventajas son:

- Carece de soporte técnico, en caso de que se presente algún problema, se puede consultar en foros especializados en la web.
- Requiere implementar una infraestructura por parte de la institución que lo adopta.
- La capacitación de los profesores corre a cargo de la institución que adopta el sistema.

Algunos ejemplos de Sistemas de Gestión del Aprendizaje de distribución libre o código abierto son:

- **ILIAS** (Integrated Learning, Information and cooperAtion System) (University of Cologne, 2004) es un sistema de gestión del aprendizaje de código abierto. Posee un ambiente de aprendizaje flexible, el cual permite la reutilización y el intercambio de materiales de instrucción entre

docentes y alumnos. Destaca su sistema de control de acceso basado en roles y la facilidad para la creación de cuestionarios.

- **ATutor** (Toronto University, 2002) es un sistema de código abierto que se utiliza para la impartición de cursos en línea. Su instalación es rápida y pueden desarrollarse temas para cambiar la apariencia de la plataforma, así como instalar nuevos módulos para ampliar las funcionalidades de la misma. Los docentes pueden crear, configurar y empaquetar cursos para su redistribución en la red y los estudiantes aprenden en un entorno adaptativo y social.
- **Sakai** (Aperio Foundation, 2016) consiste en un sistema que permite la enseñanza, aprendizaje, investigación y colaboración educativa; posee una interfaz simple, la cual puede ser personalizada por cada institución, asimismo cuenta con múltiples funciones tales como: tareas, calendarios, chat, foro de discusión, lecciones, pruebas, entre otras.
- **MOODLE** (Modular Object Oriented Distance Learning Environment) (Dougiamas y Taylor, 2003) es un sistema de gestión de cursos de aprendizaje en línea de distribución libre. Su diseño está basado del enfoque pedagógico constructivista social. Posee una arquitectura modular, lo que permite incorporar una gran diversidad de componentes y funcionalidades. Implementa una interfaz de navegador de tecnología sencilla, ligera, eficiente y compatible.

Los LMS del futuro combinarán cinco tendencias basadas en (Graduarte XXI, 2015):

- Las escuelas híbridas
- Las plataformas adaptativas
- Las estrategias de gamificación y nuevos certificados
- Las redes sociales y la creación colaborativa de aprendizajes
- La corrección automática de evaluaciones

Algunos ejemplos de estos Sistemas de Gestión del Aprendizaje son:

- **Fishtree** es un sistema que además de tener todas las funciones de los clásicos gestores de aprendizaje, tiene características que permiten al docente diseñar planes de clases con base en distintos recursos digitales, esto permite que se genere nuevas formas de impartir una clase y promueva el aprendizaje adaptativo (Fishtree, 2016).
- **Growth Engineering**, este sistema ha estado innovando mediante la gamificación de su interfaz y su uso educativo se basa en equipos, tableros de logros, retos y mucha interacción social (Growth Engineering, 2016).
- **Fidelis Education**, el objetivo de esta plataforma es reducir la deserción universitaria acompañando al alumno mediante las relaciones humanas con sus mentores o tutores. Incorpora estrategias de gamificación y está vinculado con redes sociales corporativas como LinkedIn, de esta forma obtiene que los empleadores puedan hacer recomendaciones en que campos del desarrollo los estudiantes deben mejorar y así ellos puedan

desarrollar las competencias que se requieren para el campo laboral

(Fidelis, 2016).

- **Dreambox**, este LMS combina una experiencia de aprendizaje altamente personalizada de las matemáticas con un plan de estudios riguroso para la comprensión profunda de los conceptos. Este software captura cada actividad realizada por los alumnos y puede anticipar 60 parámetros distintos de comportamiento (por ejemplo, frecuencia, tipo y velocidad de respuestas, cantidad y tipos de errores, entre otros). El programa amasa una cantidad inmensa de datos por alumno. Con esta información cambia la presentación, el tipo de clases y la secuencia siguiente en tiempo real ante cada alumno (DreamBox, 2016).
- **Smart Parrow**, este software es de la Universidad de South Wales en Australia. Ofrece una interfaz adaptada para que escuelas y docentes creen clases adaptativas, simulaciones de ejercicios y tareas inteligentes. Este LMS está especialmente direccionado a docentes, debido a que les ofrece la autoría de las clases con un software que les permite crear trayectos adaptativos. Tiene un enfoque centrado en aprender haciendo más que en la memorización (Smart Parrow, 2016).

### **2.3.1. Sistemas de gestión del aprendizaje y competencias.**

Respecto al uso de los Sistemas de Gestión del Aprendizaje para el desarrollo de competencias en los alumnos, López, Romero y Roperó (2010) realizan un análisis en el que describen las competencias generadas por medio de las herramientas de MOODLE:

Foro de trabajo en grupo: Otra posibilidad que da esta herramienta es habilitar un foro para el trabajo en grupo. De esta forma, los alumnos pueden continuar con la realización del trabajo en equipo incluso en momentos en los que no están físicamente reunidos. Esta actividad fomenta el trabajo en equipo, la cooperación y las habilidades comunicativas.

Foro: En un foro una persona (el profesor o un alumno) abre un tema y el resto de compañeros pueden añadir comentarios sobre el mismo. Hay distintas formas de utilizar esta herramienta, cada una de ellas ayuda a desarrollar distintas competencias.

Foro de noticias: En la Universidad Europea de Madrid se denomina así al foro que el profesor utiliza para avisar a los estudiantes de noticias importantes que puedan tener que ver con la próxima clase, la actividad que están realizando, actividades universitarias, etc. Tal y como está creado, los alumnos reciben un e-mail cada vez que se incluye un nuevo mensaje en el foro. Se convierte así en una especie de tablón de anuncios en el que el profesor incluye los eventos relevantes de la asignatura. Los alumnos, además de recibir la noticia vía e-mail en el momento que el profesor decide comunicarla, pueden consultarla posteriormente siempre que lo necesiten.

La utilización de este foro es más pasiva por parte de los estudiantes, por lo que es el que menos competencias permite desarrollar. Sin embargo, sí que desarrolla un sentido

de la responsabilidad, al tener que mirar el e-mail con cierta regularidad. Esto es sobretodo importante en el primer curso universitario.

Foro de dudas: Muchos profesores abren un foro para que los alumnos puedan compartir sus dudas. Los alumnos pueden de esta forma resolver las dudas de sus compañeros. El alumno que plantea su duda está desarrollando la iniciativa y el sentimiento de responsabilidad. El alumno que resuelve la duda también está desarrollando la iniciativa, y las competencias de cooperación, liderazgo y habilidades comunicativas, ya que resulta normalmente más complicado explicarse por escrito que de forma oral.

Foro de debate: En numerosas asignaturas resulta de especial interés debatir ciertos temas. Sin embargo, en ocasiones la falta de tiempo imposibilita la realización de estos debates en el aula.

Los foros de MOODLE presentan una nueva forma de debatir distintos temas, ya que los debates no son en tiempo real. Esto hace que los alumnos puedan consultar lo que han escrito sus compañeros, interiorizarlo y responder en distintos momentos del tiempo. Desarrollan así competencias generales como son la comprensión escrita, las habilidades comunicativas y la iniciativa, así como competencias específicas como la comprensión del tema en debate. En este tipo de foros se pueden debatir temas propuestos por el profesor o bien temas propuestos por los alumnos, que en muchas ocasiones les resultan más interesantes y les enriquecen más.

Wiki: Es una herramienta tecnológica que permite crear un único documento entre varios participantes. Resulta de especial utilidad para el desarrollo de contenidos de distinta naturaleza entre los alumnos de toda la clase, o entre grupos; ya que el profesor puede conocer en todo momento qué ha escrito cada alumno y evaluarlo según su participación.

Con los wikis se mejoran tanto las habilidades comunicativas, como el razonamiento crítico, ya que el desarrollo de un tema en grupo de forma colaborativa no solo implica escribir algo nuevo, sino que requiere mejorar lo que ya han escrito el resto de compañeros. Cada componente del grupo puede añadir contenidos, pero también puede eliminar contenidos de sus compañeros que no considere relevantes. Se promueve así el trabajo colaborativo, ya que el resultado final no depende del trabajo individual, sino del trabajo de todo el grupo. Más información acerca de este tema se puede encontrar en De la Cruz y otros, (2007 y 2008), que tratan del modo en el que esta herramienta permite desarrollar el aprendizaje de modo colaborativo comparándola con otras herramientas como los foros y los blogs.

Chat: Como es de conocimiento común, el Chat es una herramienta que permite la comunicación simultánea a través de internet entre dos o más personas. MOODLE pone a nuestra disposición esta herramienta para su uso en el entorno académico. Esta herramienta resulta especialmente interesante en asignaturas no presenciales ya que permite un acercamiento entre el profesor y los alumnos al permitirles comunicarse en tiempo real. Son múltiples las aplicaciones que puede tener el Chat en el ámbito de la

enseñanza universitaria. Se enumeran aquí algunas de las que se han empleado y las competencias que han permitido desarrollar con los alumnos.

Chat de dudas: Este tipo de chat permite al alumno consultar sus dudas en tiempo real. El profesor debe comprometerse a conectarse de forma periódica en momentos del tiempo determinados, para que los alumnos puedan consultar sus dudas. Puede resultar también interesante que sean los alumnos los que se comprometan en turnos rotativos a conectarse al Chat para resolver las dudas de sus compañeros, en caso de que sepan hacerlo y gestionarlas con el profesor en caso de que no sepan. No obstante, es recomendable que el profesor esté conectado en todas las sesiones para intervenir si es necesario. En esta situación el alumno desarrolla competencias generales como la comunicación escrita, la comprensión escrita, la iniciativa y la responsabilidad, así como competencias específicas derivadas de la resolución de las dudas propuestas. Es recomendable que las distintas sesiones del Chat de dudas permanezcan visibles a los alumnos para que puedan consultarlas en cualquier momento.

Chat de debate: El Chat resulta una herramienta ideal para realizar debates online en tiempo real, ya que permite a varias personas estar conectadas a la vez. Esto posibilita el desarrollo de un debate prácticamente como si los participantes estuvieran en la misma sala, para lo que deben estar conectados en el mismo momento. Es importante en esta actividad estipular unos turnos de palabra, como si de un debate presencial se tratase, para que todos los alumnos puedan participar, pero siempre que hayan leído y comprendido las aportaciones de sus compañeros. Se desarrollan aquí competencias similares a las desarrolladas en los foros de debate como son la comprensión escrita, las

habilidades comunicativas y la iniciativa, además de la comprensión e interiorización del tema en debate. Sin embargo, es un complemento perfecto al foro de debate, ya que en el chat no hay tanto tiempo para la reflexión. Por tanto, gracias a esta herramienta los alumnos comprenderán lo importante que es el desarrollo de la capacidad de comunicación escrita para hacerse entender rápidamente. Los temas a debate también aquí pueden ser propuestos por el profesor o por los estudiantes e incluso pueden ser moderados por uno u otros.

Si el debate es moderado por los estudiantes se desarrolla además la competencia de la responsabilidad. Como se puede comprobar estas utilidades son muy similares a las comentadas previamente para los foros, ya que ambas son herramientas que nos permiten la comunicación con los alumnos con la única diferencia de la temporalidad y la instantaneidad.

**Cuestionario:** Los cuestionarios son una herramienta especialmente útil para los profesores ya que una vez diseñados permiten realizar una evaluación automática de los alumnos. Podemos enfocar los cuestionarios de dos formas:

**Cuestionarios de entrenamiento:** Estos cuestionarios podrán ser resueltos por el alumno tantas veces como él considere necesario. La calificación obtenida con este tipo de cuestionarios servirá tan sólo para orientar al alumno de cuál es su conocimiento de la asignatura. Le ayudarán por tanto a desarrollar competencias generales como la responsabilidad y la planificación, pero sobre todo le ayudarán a desarrollar de forma autónoma las competencias específicas que le posibilitarán la superación con éxito de la asignatura. En los cuestionarios de entrenamiento es recomendable que se permita al

alumno realizar cada intento sobre los anteriores, permitiéndole así estudiar cuál ha sido su respuesta en el intento previo y tomar una decisión acerca de si debe cambiarla o no. Es también recomendable que entre intento e intento se permita al alumno conocer su calificación y se le dé alguna orientación sobre la respuesta seleccionada previamente.

**Cuestionarios de evaluación:** Se recomienda que en este tipo de cuestionarios sólo se permita al alumno emplear un intento y con un tiempo limitado. Este tipo de cuestionarios están más orientados a la evaluación que al desarrollo de competencias. No obstante, al estar normalmente fijados para unas fechas concretas desarrollan como cualquier prueba de evaluación la responsabilidad de los estudiantes. Los cuestionarios permiten dar rápidamente una retroalimentación a los alumnos, además de ser para ellos oportunidades de aprendizaje. Les marcan una fecha para la que tienen que haber asimilado unos contenidos, y pueden ver en el mismo momento en que terminan el cuestionario qué tal lo han entendido todo. El profesor además puede adjuntar comentarios junto con las respuestas adecuadas a cada pregunta. También se pueden utilizar los cuestionarios para realizar evaluación entre iguales, lo que permite construir conocimiento de forma conjunta. Una discusión sobre cómo hacerlo se encuentra en Zenha-Rela y Carvalho (2006).

**Glosario:** Esta herramienta permite que los alumnos vayan construyendo un glosario común de terminología relacionada con la materia. Este glosario permanecerá a su disposición a lo largo de la asignatura permitiendo que en cualquier momento puedan consultar los conceptos que necesiten. Durante la construcción del mismo, los alumnos aprenden a buscar información, extraer la que resulta relevante, comprenderla e

integrarla, desarrollando así la competencia de integración de la información.

Posteriormente deben ser capaces de redactar de forma clara, estructurada y original la información encontrada. De este modo deberán comprender el concepto en cuestión en profundidad para ser capaces de explicarlo con claridad y originalidad, desarrollando así la consecuente competencia específica. Además, deben ser consecuentes con el trabajo realizado ya que el glosario, aun siendo supervisado por el profesor, lo construyen íntegramente los alumnos y sus compañeros lo usarán posteriormente para estudiar los distintos conceptos. De este modo cada alumno debe estar seguro de que lo incluido en el glosario es correcto y además está claramente definido para que sus compañeros no tengan dudas cuando lo consulten. Se desarrolla aquí por tanto la responsabilidad.

También se puede trabajar en el glosario formando grupos pequeños que se encarguen de incluir los términos de forma conjunta, desarrollando así la competencia de trabajo en equipo. Esta competencia se desarrollará en cualquier caso, ya que, como el resto de los compañeros tendrán el glosario como referencia de consulta deben exigir a sus compañeros que el trabajo realizado tenga la calidad suficiente para ser consultado de forma adecuada. Deben por tanto aprender a ser críticos con el trabajo propio y con el de los demás.

Consultas: MOODLE también permite realizar consultas a los alumnos. En estas consultas, el profesor plantea una cuestión que sea de interés general y los alumnos pueden elegir la respuesta que mejor se adapte a sus necesidades por medio de una votación. En todas las preguntas las posibles respuestas deben ser cerradas. De esta manera, se pueden realizar votaciones sobre el funcionamiento de la asignatura. Esto hará

que los alumnos se impliquen más en el desarrollo de las clases, al entender que su opinión es tomada en cuenta por el profesor, de cara a modificar en cada caso lo que se considere oportuno. Esto facilita enormemente la comunicación entre profesor y alumnos, sobre todo en aquellos casos en los que los alumnos o, bien por timidez o por otros motivos no se atreven a hacer saber sus inquietudes al profesor. Con las consultas se desarrolla la responsabilidad de los alumnos, pues si sus opiniones son tomadas en cuenta deben meditarlas cuidadosamente antes de ser puestas en común. Además, se fomenta la iniciativa, ya que al ver los alumnos que sus opiniones son escuchadas tomarán la iniciativa de comentar con el profesor otras sugerencias que pueden resultar de gran interés para el seguimiento y enriquecimiento de la asignatura.

Documentación: Si hacemos una reflexión rápida sobre la utilidad de los campus virtuales en el ámbito de la docencia la primera idea es su utilización para compartir documentación con los alumnos. No obstante, después de lo ya enumerado se puede comprobar que aportan muchas otras funcionalidades. En cualquier caso, sí se ha de tener presente que esta es una de las principales utilidades y probablemente la más extendida en uso. El profesor siempre debe tener una planificación de cómo será el desarrollo de la asignatura desde el principio, pero MOODLE le hará estructurarla de una forma más visible para el alumno. Esto hace que este aprenda a planificarse mejor, ya que conoce la estructura de la asignatura casi desde su comienzo, teniendo una idea clara de lo que va a aprender y en qué momento. Además hace que los estudiantes tengan la responsabilidad de anticiparse a las explicaciones del profesor en los casos que resulte adecuado para una mayor comprensión de los distintos temas.

Agenda: En el campus virtual hay un calendario donde los profesores pueden poner las fechas de las actividades más importantes de la asignatura. También incorpora de forma automática los eventos que considera relevantes como la apertura y cierre de cuestionarios, y tareas, además de avisar de las actividades que se celebran en la universidad, dando así a alumnos y profesores una visión de global que les facilita la gestión de su tiempo. Esta agenda desarrolla la competencia de responsabilidad, al permitir a los alumnos que puedan organizar su tiempo entre las distintas asignaturas, y actividades que tienen que realizar. Asimismo, les permite planificarse y organizar su tiempo de la manera que consideren más adecuada, ya que les proporciona una visión general de las actividades más importantes que tendrán lugar, no sólo en la asignatura sino también en la universidad.

### **2.3.2 Diseño instruccional en entornos virtuales de aprendizaje.**

Uno de los componentes fundamentales del e-learning, es el diseño instruccional. El diseño instruccional ha pasado por varias etapas de desarrollo en su enfoque y práctica entre las cuales se pueden distinguir claramente modelos de desarrollo instruccional de cuatro generaciones (Polo, 2001):

La primera generación (1960): se basó en el enfoque conductista. Se formula linealmente el desarrollo de la instrucción. Se caracteriza por ser sistémico, es decir, procede paso a paso y prescribe los métodos específicos y programados, los cuales han sido centrados en el conocimiento y destrezas de tipo académico, así como en la formulación de objetivos de aprendizajes observables y secuenciales.

La segunda generación (1970), está fundamentada en los macro-procesos, es decir, en sistemas más abiertos, en donde se toman en cuenta aspectos internos y externos de la instrucción, con prescripciones pedagógicas para seleccionar estrategias instruccionales y secuencias transaccionales, que permiten una mayor participación cognitiva del estudiante.

Los diseños instruccionales de la tercera generación (1980) han sido llamados también diseños instruccionales cognitivos, ya que sus estrategias son heurísticas. Los contenidos pueden ser planteados como tácitos y los conocimientos deben ser de tipo conceptual, factual y procedimental, basados en la práctica y en la resolución de problemas.

La cuarta generación (1990), está fundamentada sobre la primicia de que existen diversos mundos epistemológicos, lo que la diferencia de los anteriores. Se caracteriza por sustentarse en las teorías constructivistas, la del caos, la de los sistemas, lo cual da como resultado un modelo heurístico. Además, está centrado en el proceso de aprendizaje y no en los contenidos específicos. Este modelo permite al diseñador combinar los diferentes materiales con las actividades propuestas, para conducir al estudiante a desarrollar habilidades en la creación de interpretaciones por sí mismo, y manipular situaciones hasta que las asuma como parte del aprendizaje, es decir, la motivación no es sólo un factor externo sino interno.

De acuerdo con Gordillo y Chávez (2007), el diseño instruccional es el proceso en donde se analizan, organizan y presentan objetivos, información, actividades, métodos,

medios y el proceso de evaluación, que al conjugarse entre sí conforman el contenido de un curso con miras a generar experiencias satisfactorias de aprendizaje.

Lo anterior, implica que dentro del contexto de los entornos virtuales de aprendizaje, el contenido de un curso debe diseñarse específicamente para utilizarse en un ambiente virtual interactivo, que permita: el acceso a la información de manera compartida; diversas formas de comunicación asincrónica y sincrónica; integración de diferentes tipos de información audiovisual (videoclips, animaciones, efectos sonoros, música, fotografía, enlaces a otras páginas, entre otros); facilitar contextos de aprendizaje; proporcionar recursos humanos y electrónicos; herramientas o medios para introducir y manipular tanto las ideas como los recursos y establecer apoyos a los procesos que ayuden a la tarea individual del aprendizaje (Tobón 2007).

Existen algunas iniciativas para generar herramientas para el diseño instruccional en la educación superior, tales como el Generador de Diseño Instruccional propuesto por Francesa, Espinoza y Chacón (2012), mismo que se basa en un LMS implementado en el Tecnológico de Costa Rica.

Urquijo y Tapia (2007) mencionan el desarrollo de la herramienta “Temachitani” (en náhuatl, “el que enseña”) implementada en la Universidad de La Salle la cual es un entorno virtual de aprendizaje diseñado a partir de las características pedagógicas del diseño instruccional.

Estrada por su parte menciona que las aplicaciones de software educativo han evolucionado poco en lo relacionado con la fase del diseño instruccional y propone un

modelo de datos para describir los elementos que intervienen en el diseño de cursos virtuales e ilustrar las relaciones entre tales elementos. (Estrada 2016).

## 2.4. Gestión del conocimiento.

El objetivo de esta sección es describir los diferentes conceptos y procesos involucrados en el descubrimiento y gestión del conocimiento.

### 2.4.1 Definiciones y taxonomía.

El descubrimiento de conocimiento implica diversos conceptos con los que está relacionado, tales como los Datos, la Información, la Gestión de la información y la gestión del conocimiento. Los datos consisten en hechos, imágenes, o sonidos. Cuando se combinan con la interpretación y el significado, se convierte en información. Por lo tanto, la información consiste en datos que han sido formateados, filtrados, y resumidos (Chen, 2001). El proceso de la transformación de datos en información se representa en el siguiente esquema (Itmazi, 2005):



*Figura 3. Proceso de transformación de datos en información.*

Una vez establecidos los conceptos datos e información, se puede definir del término de Gestión de la Información (GI), como “el proceso que utiliza herramientas y técnicas para capturar, recoger, analizar, organizar, manipular y recuperar la información. De la misma manera, controla y explota los recursos de la información de una organización” (Itmazi, 2005).

Por otro lado, la Gestión del Conocimiento (GC), se define como una disciplina emergente cuyo objetivo es generar, compartir y utilizar el conocimiento tácito y explícito existente en un determinado espacio para dar respuesta a las necesidades de los individuos y de las comunidades en su desarrollo (Alavi et al., 2001).

Finalmente, el Descubrimiento del Conocimiento en Bases de Datos (en inglés, Knowledge Discovery in Databases, KDD), se define como “el proceso no trivial de identificar patrones válidos, novedosos y potencialmente útiles y en última instancia, comprensible a partir de los datos” (Fayyad et al., 1996).

El KDD es un proceso iterativo e interactivo y de acuerdo con Hernández (Hernández et al., 2004) su taxonomía se puede organizar en cinco fases (ver Figura 2):

**Integración y recopilación.** En esta actividad se determinan las fuentes de información que pueden ser útiles y dónde conseguirlas. Posteriormente, se transforman los datos a un formato común. Dado que los datos provienen de diferentes fuentes, pueden contener valores erróneos o faltantes.

**Selección, limpieza y transformación.** Se eliminan o corrigen los datos incorrectos y se decide la estrategia a seguir con los datos incompletos. Además, se proyectan los datos para conseguir únicamente aquellas variables o atributos que van a ser relevantes, con el objetivo de hacer más fácil la tarea propia de minería y para que los resultados de la misma sean más útiles.

**Minería de datos.** Se decide cuál es la tarea por realizar (clasificar, agrupar, entre otras) y se elige el método a utilizar.

**Evaluación e interpretación.** Los expertos evalúan y se analizan los patrones, y si es necesario se vuelve a las fases anteriores para una nueva iteración. Esto incluye resolver posibles conflictos con el conocimiento que se disponía anteriormente.

**Difusión y uso.** Se hace uso del nuevo conocimiento y se hace partícipe de él a todos los posibles usuarios. Este conocimiento se suele utilizar en procesos de toma de decisiones.

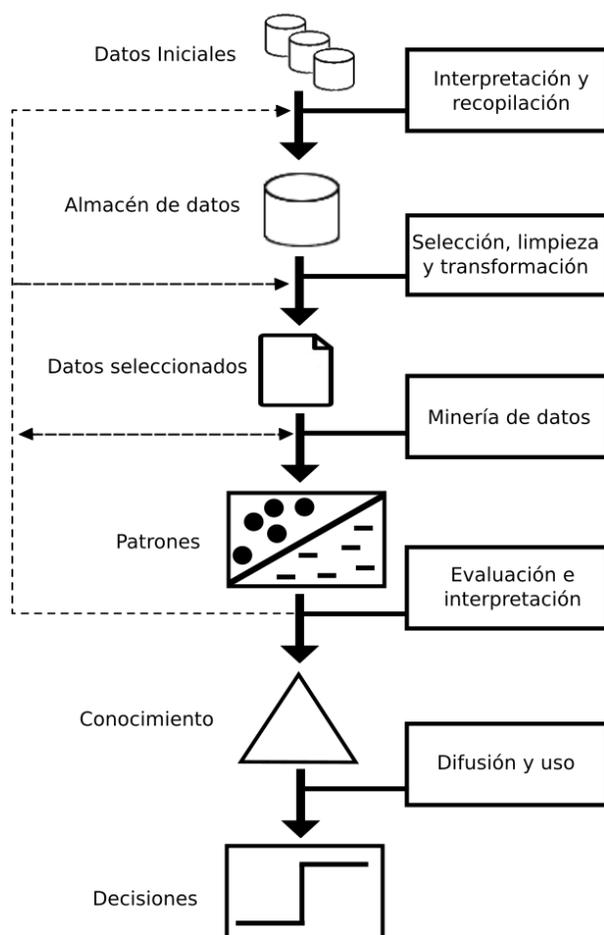


Figura 4. Proceso de transformación de datos en información.

## 2.5. Minería de datos

Como se observa en la Figura 2 la minería de datos es una etapa crucial de las técnicas incluidas en el proceso KDD. Se define como “el proceso de extraer conocimiento útil y comprensible, previamente desconocido, desde grandes cantidades de datos almacenados en distintos formatos” (Fayyad et al., 1996).

La minería de datos trabaja con grandes volúmenes de datos, procedentes en su mayoría de sistemas de información, con los problemas que ello conlleva (ruido, datos ausentes, intratabilidad, volatilidad de los datos, entre otros), y aplica técnicas adecuadas para analizar estos datos y extraer conocimiento novedoso y útil (Hernández et al., 2004).

De acuerdo con (Hernández et al., 2004), los modelos de aprendizaje empleados en la minería de datos pueden dividirse en dos categorías principales: predictivos y descriptivos.

Cada una de estas categorías contiene una taxonomía la cual se describe (ver Figura 4):

**Modelos predictivos o aprendizaje supervisado.** Predicen el valor del atributo de un conjunto de datos a partir de información previamente conocida. El objetivo de este modelo es proporcionar características que describan las relaciones entre los datos. Los algoritmos aplicados en los modelos predictivos se catalogan en:

**Clasificación.** Son utilizados para clasificar datos, con la finalidad de predecir clases de objetos cuyas categorizaciones no se han definido.

**Regresión.** Predicen una o más variables continuas, como por ejemplo las pérdidas o los beneficios, basándose en otros atributos del conjunto de datos.

Predicción. Son aquellos que pueden ser usados para predecir tendencias de los valores en los datos o de las clases.

**Modelos descriptivos o aprendizaje no supervisado.** Descubren patrones y tendencias en los datos. Sirven para llevar a cabo acciones y obtener un beneficio o conocimiento de ellas. Estos algoritmos se clasifican en:

Agrupamiento. Son aquellos que no requieren una clasificación predefinida para particionar los datos obteniendo el conocimiento de acuerdo con las características de los mismos.

Asociación. Se basan en el descubrimiento de reglas de asociación que muestran condiciones en los valores de los atributos que ocurren simultáneamente de forma frecuente en un determinado conjunto de datos.

Correlación y dependencias. Establecen un patrón en el que uno o más atributos determinan el valor de otro. Su función principal es predecir diferentes valores de los datos, como ejemplo: ganancias, ventas, tasas, entre otros.

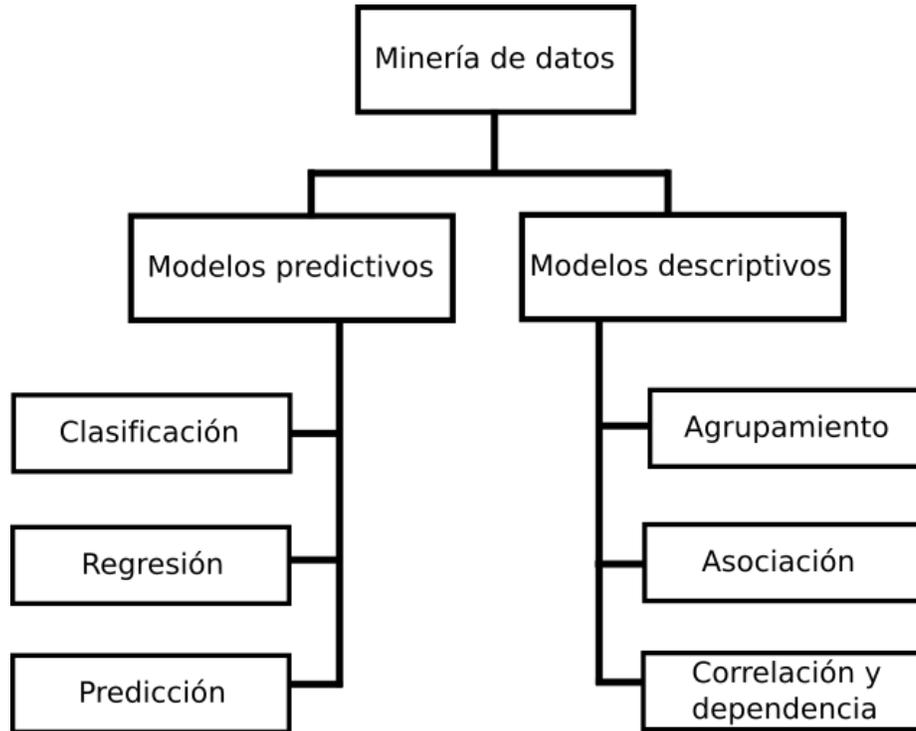


Figura 5. Taxonomía de los modelos utilizados en minería de datos.

En la minería de datos existen multitud de técnicas que pueden resolver una gran diversidad de problemas. A continuación, se presenta un resumen de las técnicas más populares:

**Algebraicas y estadísticas.** Se basan, generalmente, en expresar modelos y patrones mediante fórmulas algebraicas, funciones lineales, funciones no lineales, distribuciones o valores agregados estadísticos tales como medias, varianzas, correlaciones, entre otras. Algunos de los algoritmos más conocidos dentro de este grupo de técnicas son la regresión lineal (Edwards, 1976) y la regresión logística (Hosmer, 2000).

**Redes bayesianas.** Se basan en estimar la probabilidad de pertenencia (a una clase o grupo), mediante la estimación de las probabilidades condicionales inversas o a

priori, utilizando el teorema de Bayes. Poseen varias ventajas, una de ellas es que pueden tratar muchos atributos y son muy robustas al ruido que pueden provocar los datos. Por el contrario, la expresividad es limitada y depende de la discretización. Algunos algoritmos populares son EM (Dempster et al., 1977) y Naive Bayes (H. Zhang, 2004).

**Algoritmos evolutivos.** Son métodos sistemáticos para la resolución de problemas de búsqueda y optimización, a los cuales se aplican los mismos métodos de la evolución biológica, tales como: selección basada en la población, reproducción sexual y mutación (Eiben y Smith, 2003).

**Conteos de frecuencias y tablas de contingencia.** Se basan en contar la frecuencia en la que dos o más sucesos se presenten conjuntamente. Ejemplos de estos algoritmos son Apriori (Agrawal et al., 1993) y Predictive A priori (Scheffer, 2004).

**Árboles de decisión.** Está técnica se encuentra dentro de una metodología de aprendizaje supervisado. Su representación es en forma de árbol, en donde cada nodo es una decisión, los cuales a su vez generan reglas para la clasificación de un conjunto de datos. Entre sus ventajas se encuentran: que admiten atributos discretos y continuos, también se desempeñan de forma eficiente con los atributos no significativos. Entre sus desventajas, se encuentran la sobreadaptación y el ruido generado por los datos. Algunos ejemplos de estos algoritmos son ID3 (Quinlan, 2007), J48 (Ye, 2006) y C4.5 (Quinlan, 1993).

**Relacionales, declarativas y estructurales.** La característica principal de este conjunto de técnicas es que representan los modelos mediante lenguajes declarativos, como los lenguajes lógicos, funcionales o lógico-funcionales. Las técnicas de ILP

(programación lógica inductiva) son las más representativas y las que han dado nombre a un conjunto de técnicas denominadas minería de datos relacional.

**Redes neuronales artificiales.** Se utilizan de manera frecuente para detectar categorías comunes en los datos, debido a que detectan y aprenden complejos patrones de comportamiento. Sus principales ventajas son: cuando están bien ajustadas, obtienen precisiones muy altas y se puede aplicar tanto en modelos predictivos como descriptivos. Su principal desventaja son los datos incompletos. Algunos ejemplos son las redes de kohonen (Kohonen, 1988) y el modelo de Retro-Propagación (Rumelhart et al., 1986).

**Basadas en núcleo y máquinas de soporte vectorial.** Se trata de técnicas que intentan maximizar el margen entre los grupos o las clases formadas. Existen muchas variantes, dependiendo del núcleo utilizado y la manera de trabajar con el margen.

**Estocásticas y borrosas.** Este conjunto de técnicas contiene características que se describen por separado. Por una parte, las técnicas estocásticas se basan en las teorías del análisis de probabilidades. Contienen datos con alguna función de densidad de probabilidad asociada. Por otra parte, las técnicas borrosas, aunque también se basan en la presunción de un conocimiento impreciso, son distintas de las estocásticas. En vez de tener una función de densidad de probabilidad asociada, se caracterizan por una función de pertenencia que expresa el grado en que un elemento pertenece a un conjunto. Estas técnicas junto con las redes neuronales y los algoritmos evolutivos, forman lo que se denomina computación flexible.

**Basadas en casos, en densidad o distancia.** Son métodos que se basan en distancias al resto de elementos, ya sea directamente, como los vecinos más próximos o

mediante la estimación de funciones de densidad. Algunos algoritmos conocidos son el K-medias (Macqueen, 1967) y COBWEB (Fisher, 1987).

Además de todo lo anterior existen diversas combinaciones que dificultan aún más realizar una taxonomía razonable que contemple a todos los grupos de técnicas.

Por otro lado, existe una variedad de paquetes software para realizar minería de datos. Cada uno de ellos posee características apropiadas para realizar determinadas tareas o para analizar cierto tipo de datos. A continuación, se describen algunos ejemplos:

- **WEKA** (Waikato Environment for Knowledge Analysis) (Witten, 2011), es una herramienta visual de libre distribución desarrollada por los investigadores de la Universidad de Waikato en Nueva Zelanda. Está implementado en el lenguaje de programación Java y opera en diferentes entornos de sistemas operativos. Algunas de sus principales características son: acceso a datos, preprocesado de datos, modelos de aprendizaje y contiene cuatro entornos.
- **KEEL** (Knowledge Extraction based on Evolutionary Learning) (Alcalá-Fdez et al., 2008), fue desarrollado por 5 grupos de Universidades españolas, permite utilizar y construir diferentes modelos para minería de datos y tiene como característica importante la inclusión de una librería de algoritmos de aprendizaje evolutivo con código abierto en Java. Las características principales son: algoritmos de preprocesamiento, biblioteca de algoritmos de extracción de conocimiento, biblioteca de herramientas estadísticas para el análisis de los algoritmos y aplicación para su uso vía web; además contiene cuatro entornos.

- **DBMiner** (DBMiner Technology Inc., 2003), es un software de distribución libre desarrollado por la Universidad de Simon Fraser, Canadá. Está concebido para la extracción del conocimiento en bases de datos relacionales, almacenes de datos y web, su arquitectura de diseño incorpora OLAP (online analytic processing) y OLAM (online analytic mining).
- **RapidMiner** (Universidad de Dortmund, 2001). Este sistema antes era conocido como YALE (Yet Another Learning Environment), está desarrollado en Java y es de código abierto. Cuenta con una interfaz gráfica fácil de utilizar y además integra un mecanismo sencillo para desarrollar extensiones que hacen posible integrar nuevos operadores y con ello adaptar el software para los requerimientos personales.

## 2.6. Minería de datos en educación

La Minería de Datos en Educación o Minería de Datos Educativa (Educational Data Mining, EDM) es la aplicación de técnicas de minería de datos a información generada en los entornos educativos. La EDM se define como “el proceso de transformar los datos en bruto recopilados por los sistemas de enseñanza en información útil que pueda utilizarse para tomar decisiones informadas y responder preguntas de investigación” (Heiner et al., 2006).

La aplicación de minería de datos en sistemas educativos es un ciclo iterativo de formación de hipótesis, pruebas y refinamiento. El conocimiento descubierto, una vez filtrado, sirve de guía, facilita y mejora el proceso de aprendizaje a través de la toma de

decisiones. Como se observa en la Figura 2.6, los educadores, investigadores y responsables académicos son los encargados de diseñar, planificar, crear y mantener los sistemas educacionales y los alumnos usan e interactúan con él (Romero y Ventura, 2007).

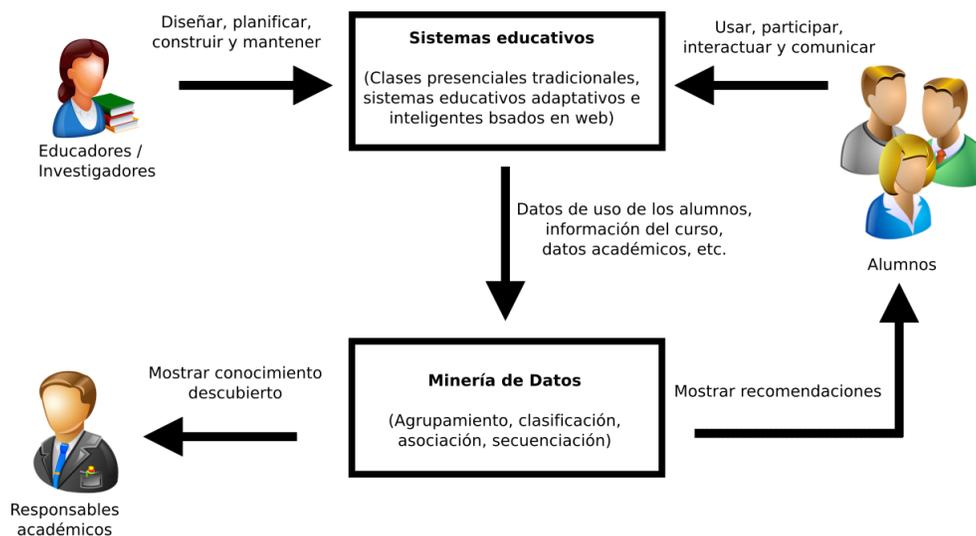


Figura 6. Interacción de alumnos, académicos e investigadores con los sistemas educativos y el proceso de minería de datos (tomado de Romero y Ventura, 2007).

La aplicación de técnicas de EDM se puede ver desde distintos puntos de vista dependiendo del tipo de usuario (Romero y Ventura, 2010; Romero y Ventura, 2013):

- **Del estudiante.** El objetivo es ayudar o realizar recomendaciones a los alumnos durante su interacción con el sistema de e-learning para mejorar su aprendizaje. Sus principales aplicaciones son: sugerir buenas experiencias de aprendizaje a los estudiantes, adaptar el curso según el progreso del aprendiz, ayudar a los estudiantes dando sugerencias y atajos, recomendar caminos más cortos y personalizados, entre otros.

- **Del profesor.** El objetivo es apoyar a los profesores de los sistemas de e-learning para que puedan mejorar el funcionamiento o rendimiento de estos sistemas a partir de la información generada por los alumnos al utilizar los cursos. Sus principales aplicaciones son: obtener una mayor realimentación de la enseñanza, conocer más sobre cómo los estudiantes aprenden en la web, evaluar a los estudiantes por sus patrones de navegación, reestructurar los contenidos en el sitio web para personalizar los cursos, clasificar a los estudiantes en grupos, entre otros.
- **De los investigadores.** El objetivo es comparar técnicas de minería de datos que permitan recomendar la más útil para una tarea específica educativa o problema, para evaluar la efectividad del aprendizaje utilizando diferentes configuraciones y métodos, y así sucesivamente.
- **De las instituciones educativas.** El objetivo es obtener información útil que permita mejorar la eficiencia del sitio y adaptarlo a la conducta de los usuarios (tamaño óptimo del servidor, distribución del tráfico de red, entre otros). Otros objetivos son adquirir medidas que permitan organizar mejor los recursos institucionales (humanos y materiales) y determinar la eficiencia de los programas educativos del aprendizaje a distancia soportado por ordenador.

A continuación se describen algunos ejemplos de trabajos donde se han aplicado con éxito:

Zhang (Zhang et al., 2008), propone un Sistema de Instrucción Personalizada de Recomendación (PIRS, en inglés), el cual está diseñado para el aprendizaje basado en web. Este sistema reconoce los diferentes patrones de estilo de aprendizaje y hábitos de uso de internet a través de pruebas aplicadas a los estudiantes. Para ello, implementan reglas de asociación a través del algoritmo Apriori a sus registros de navegación web.

Un enfoque similar es el propuesto por Ba-Omar (Ba-omar et al., 2007), es un sistema centrado en la identificación de patrones de aprendizaje de los estudiantes y la secuencia de elección de los recursos educativos (audio, texto, entre otros) en relación con sus estilos de aprendizaje. Para detectar estos patrones utilizaron el algoritmo Apriori.

Romero (Romero et al., 2009), propone una arquitectura avanzada cuyo objetivo es realizar recomendaciones a los estudiantes los sitios o páginas web más apropiados cuando visitan un sistema educativo adaptativo basado en web. Estas recomendaciones se basan de las reglas descubiertas por diversos algoritmos de minería web.

La propuesta de Rištofič (Rištofič, 2005), se enfoca a la recomendación de lecciones (objetos de aprendizaje o conceptos) que los estudiantes deben estudiar la próxima vez que utilicen un sistema educativo adaptativo basado en web. Para descubrir los patrones de comportamiento se implementaron reglas de asociación.

García (García et al., 2007), propone un sistema de recomendación basado en técnicas de minería de datos que ayude al profesor a detectar posibles problemas en la estructura y contenidos de los cursos, basado en la propia información que brindan los datos de utilización de los cursos por los alumnos.

La propuesta desarrollada por Wang (Wang y Shao, 2004), establece un modelo de recomendación para estudiantes, el cual está basado en intervalos de tiempo específico de las sesiones de navegación realizadas por cada usuario. Posteriormente, se integran grupos de usuarios con perfiles de navegación similares y se ejecutan técnicas de agrupamiento y reglas de asociación.

Vialardi (Vialardi et al., 2009), propone un sistema para predecir qué tan conveniente es para un estudiante matricularse a un curso específico. Para ello, se utilizan los resultados de estudiantes con un perfil similar que han llevado el curso. Para descubrir los patrones de comportamiento se implementó el algoritmo C4.5.

Finalmente, la propuesta de Guo (Guo y Zhang, 2009), es un sistema de recomendación orientado al aprendizaje personalizado y adaptativo de los estudiantes basado en el desempeño de las actividades realizadas en el pasado. Para ello, utilizan redes neuronales y árboles de decisión.

## **2.7. Agentes conversacionales.**

Los agentes conversacionales (en inglés *Conversational Agent*, CA o chatbot) son programas de software que permiten a las personas interactuar con sistemas informáticos utilizando diálogos de lenguaje natural. Los agentes conversacionales se han desarrollado para servir a múltiples funciones pedagógicas, como tutores, entrenadores o compañeros de aprendizaje (Haake y Gulz, 2009). En sistemas expertos, como tutores (Greaser et al. 2005), las interfaces de CA son intuitivas y útiles para involucrar a los usuarios en la discusión, como lo demuestra (Tegos, Demetriadis y Karakostas, 2015). Los CA pueden

agregar capacidades de diálogo natural a ITS pero no se usan con frecuencia ya que su desarrollo es complejo y requiere mucho tiempo, además de requerir experiencia en la creación de scripts de diálogo (O'shea, Bandar y Crockett, 2011).

Los agentes conversacionales se implementan utilizando dos enfoques principales:

- **Coincidencia de Patrones:** Estos agentes buscan palabras clave y expresiones regulares en la entrada del usuario para tomar acciones según la entrada.
- **Análisis Semántico:** intentan comprender el significado de la entrada del usuario utilizando técnicas de análisis semántico, como el análisis de semántica latente (LSA) y la asignación de Dirichlet latente (LDA).

La mayoría de los CA de coincidencia de patrones se desarrollan utilizando pares estímulo-respuesta ya que son eficaces para tratar expresiones sintácticas mal formadas, sin embargo, el desarrollo de AC de coincidencia de patrones a menudo implica definir el alcance del chatbot y la creación de guiones especializados. Otra consideración para estos CA es el hecho de que es difícil agregar nuevos conocimientos al sistema debido a que requieren la modificación de scripts por parte de un experto para adaptarlo a la tecnología de scripting del sistema, generalmente AIML (Artificial Intelligence Markup Language).

En los CA de análisis semántico latente, las palabras se representan como vectores y la similitud entre los conceptos se mide de acuerdo con la distancia entre los vectores. Este enfoque tiene dificultades para lidiar con un lenguaje deficiente o fallas

tipográficas. Además, si se necesita añadir conocimiento nuevo, el corpus de palabras debe ser aumentado y recalculado manualmente.

## **2.8. Trabajos relacionados**

Los LMS han evolucionado en sus funcionalidades y prestaciones hasta convertirse en plataformas complejas que soportan el ciclo completo del proceso de enseñanza y aprendizaje pues disponen de funcionalidades que apoyan la gestión de cursos, usuarios, grupos, calificaciones, etc. y también permiten gestión de contenidos (García-Peñalvo y Seoane-Pardo, 2015).

En los últimos años se ha manifestado un interés creciente en dotar a los LMS de capacidades de aprendizaje adaptativo y en particular en entornos LMS de código abierto como el MOODLE (Lerís-López, Vea-Muniesa y Velamazán-Gimeno, 2015).

Lerís-López, Vea-Muniesa y Velamazán-Gimeno (2015) hacen una revisión de las características adaptativas de MOODLE en su versión 2.4.

En su trabajo mencionan que las capacidades adaptativas de MOODLE aparecen en dos direcciones principales: intra-actividad e inter-actividad.

La primera significa que la adaptación al usuario es una posibilidad de la configuración interna de la actividad, mientras que la adaptividad inter-actividades supone el establecimiento de relaciones de dependencia entre diferentes actividades. Ejemplos de intra-actividad son los módulos de lecciones, cuestionarios, foros y SCORM (acrónimo de Sharable Content Object Reference Model).

Ejemplos de inter-actividad son las capacidades para adaptar la navegación y presentador de contenidos en función de las acciones del usuario, fechas, perfil del usuario, desempeño de otras actividades del mismo curso. Estas características se pueden configurar en el apartado de restricciones de acceso del formulario de ajustes de cada recurso o actividad.

Si bien es cierto que MOODLE ya tiene ciertas características de adaptabilidad, éstas no son automatizadas ni contienen el ciclo completo de un ITS.

Serce, Alpaslan y Jain (2008) presentan un Sistema Inteligente de Aprendizaje Adaptativo (Adaptive Intelligent Learning System AILS) el cual es desarrollado con el objetivo de ser utilizado con cualquier LMS.

La propuesta está implementada con una arquitectura de multi agentes JADE. La aplicación de la propuesta se realiza por medio de un LMS “dummy”.

La aplicación está diseñada para ser montada sobre un LMS, lo cual consiguen por medio de un agente dedicado únicamente a la comunicación entre AILS y el LMS.

Las funciones principales de adaptabilidad en el sistema pueden ser sumarizadas como modelado del usuario, presentación de materiales instruccionales adaptados, búsqueda de palabras clave y comunicación con otros sistemas.

El sistema fue diseñado con los siguientes componentes:

- Modelo del usuario.
- Instrucción adaptada.
- Comunicación con otros AILS.
- Integración con el LMS.

- Búsqueda de objetos de aprendizaje.

En el trabajo de los autores se implementó un demo que permite hacer la búsqueda de objetos de aprendizaje y regresa notas de lectura adaptadas al perfil del usuario.

La implementación del AILS resulta interesante sin embargo carece de la integración a un entorno LMS real y la implementación de los servicios de adaptividad es específica, por lo cual tienen que ser rehechos para adaptarse a nuevas plataformas LMS.

Moisa (2013) propone un nuevo modelo para el desarrollo de LMS adaptativos. En el trabajo de Moisa se presentan los estándares actuales utilizados en e-learning y describe los elementos que pueden ser utilizados para crear un LMS adaptativo: modos de control de secuencia, reglas de secuencia, controles de navegación, registros de aprendizaje.

El modelo se basa en algoritmos de inteligencia artificial que analizan la información capturada en el sistema y crea un camino de navegación adaptativo a través del contenido de aprendizaje contenido en el sistema, permitiendo una experiencia personalizada para cada usuario.

La arquitectura propuesta utiliza SCORM para compartir objetos entre plugins y con otros sistemas, por lo que debe de ser capaz de usar tecnologías XML. La API propuesta utiliza tecnología JSON y métodos RESTful HTTP.

El trabajo realizado explora las capacidades de los estándares actuales para producir un LMS adaptativo pero requiere del desarrollo de nuevos LMS compatibles con

el modelo propuesto. Carece de adaptación con los LMS utilizados actualmente y hace falta experimentación para comprobar los resultados de la propuesta.

Moodie y Kunz (2003) discuten los componentes necesarios para crear un Sistema de Gestión del Conocimiento Inteligente (Intelligent Learning Management System iLMS). Los autores mencionan que es necesario que los LMS que se encuentran actualmente en uso y en desarrollo pueden utilizar técnicas de AI para soportar la personalización de contenidos y mejorar el soporte a los objetivos de aprendizaje para los cuales han sido diseñados.

En el trabajo se hace mención de cuatro componentes claves para la implementación de un iLMS:

- Una caja de herramientas pedagógicas reflejada en un conjunto de plantillas para dar soporte a las actividades educacionales.
- Una librería de objetos de aprendizaje que corresponda a la colección de recursos de aprendizaje.
- Un agente que trabaje en colaboración con un profesor para ayudar a llenar los detalles de las plantillas con los objetos de aprendizaje adecuados.
- Mecanismos inteligentes que añadan riqueza al ambiente de aprendizaje colaborativo tanto para profesores como estudiantes.

Los autores hacen una revisión de las dificultades y retos asociados a la implementación de los iLMS. El trabajo realizado sirve para sentar las bases en el desarrollo desde cero de un iLMS, sin embargo no mencionan de qué manera se pueden

utilizar estos componentes para integrarlos en los LMS existentes. Los componentes identificados y propuestos por los autores son implementaciones concretas de los módulos que conforman la arquitectura básica de los ITS.

Como vemos a partir de la revisión de la literatura de los trabajos relacionados con el aprendizaje adaptativo en los LMS, las investigaciones giran alrededor de propuestas de módulos, marcos de trabajo y tecnologías existentes para crearlos, sin embargo el enfoque multidisciplinario requerido para el desarrollo e integración de un ITS en un LMS existente parece ser una barrera importante, a partir de lo cual el trabajo realizado en ésta investigación cobra relevancia al ofrecer una solución concreta y utilizable en entornos reales en los cuales se están utilizando LMS para las modalidades de educación en línea y blended learning.

Otras propuestas relacionadas a esta investigación analizan la implementación de sistemas de tutoría inteligente conversacionales o CITS (Conversational Intelligent Tutoring Systems).

Los CITS son una extensión de los sistemas ITS convencionales que permiten la interacción del usuario con el sistema utilizando lenguaje natural en lugar de menús (Latham et. al 2012). Los CITS pueden contribuir al involucramiento de los usuarios utilizando una interfaz de diálogo. Existen algunas propuestas interesantes de CITS para ayudar en el proceso de enseñanza y aprendizaje de diferentes maneras, tales como los secuenciadores curriculares (Latham et. al 2012, Mikic et al. 2012, Pilato et. al 2008, Mikic et. al, 2009), los agentes de aprendizaje colaborativo (Tegos et. al 2015) y los gestores de autoevaluación (Kerly, Ellis y Bull 2008).

Otro conjunto de investigación presenta el uso de CITS implementado usando un ITS basado en restricciones (Weerasinghe, 2009), diálogo adaptativo conducido por expectativas y conceptos erróneos (Graesser, 2016), agentes basados en creencias, deseos e intenciones (Mikic et. al 2012) e ITS híbridos (Latham et. al 2014) como alternativas a los enfoques clásicos.

Los CITS se han utilizado con éxito para aumentar la ganancia de aprendizaje en una amplia gama de dominios tales como: programación, física, aprendizaje multimedia y aprendizaje de idiomas. A pesar de esto, los sistemas ITS no han explorado si son adecuados para ayudar a los docentes en entornos de aprendizaje electrónico. En el presente trabajo se presenta un CITS híbrido para ayudar a los profesores en el uso de las herramientas de MOODLE.

### Capítulo 3. Metodología y propuesta.

En el presente capítulo se abordan los aspectos metodológicos relevantes al presente trabajo de investigación, así como la propuesta del sistema MITS (MOODLE Intelligenet Tutoring System) para asistir a los profesores en el uso de las herramientas del sistema MOODLE, su arquitectura e implementación técnica.

#### 3.1. Metodología de trabajo.

El ITS utiliza reglas de conocimiento extraídas de la interacción de los profesores con la plataforma LMS:

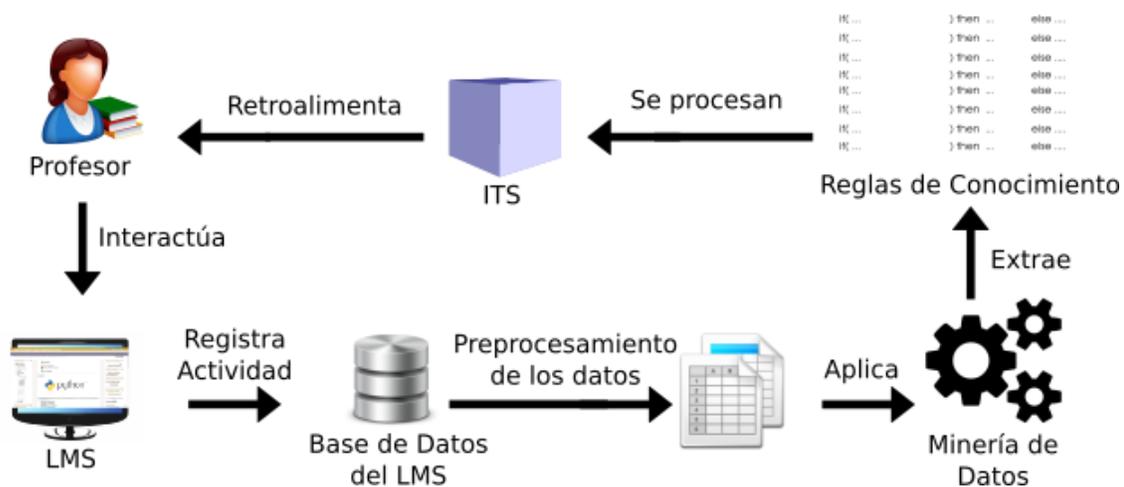


Figura 7. Esquema de extracción de conocimiento e interacción del ITS con el profesor.

Como se observa en la figura 7 el profesor interactúa con sus cursos en línea a través de un LMS, el cual registra los datos del uso de las diferentes herramientas del sistema en una base de datos. A partir de la información almacenada en dicha base de

datos se generan tablas de información en las que se tienen datos relacionados a la actividad del profesor en las diferentes herramientas del LMS. Las tablas antes mencionadas pasan por un proceso de preprocesamiento de la información y minería de datos, para extraer los patrones de comportamiento del profesor los cuales se convierten en reglas de conocimiento que son proporcionadas al tutor inteligente, el cual adapta su comportamiento tomando como base las reglas generadas y el perfil del usuario para ofrecer retroalimentación al profesor. La nueva actividad del profesor se registra en el LMS, desencadenando todo el proceso de nuevo. Éste es un ciclo continuo por medio del cual se puede personalizar las necesidades formativas de cada profesor.

El ITS se enfocará en la retroalimentación al usuario acerca del uso de las herramientas del LMS mediante un enfoque híbrido utilizando secuencia de currículo al permitir que el profesor pregunte acerca de las herramientas de su interés y tutoría basada en restricciones para brindar recomendaciones de herramientas al profesor de acuerdo con su perfil y los datos extraídos a partir del uso de la plataforma realizando recomendaciones personalizadas de acuerdo a los perfiles extraídos.

El ITS estará conformado por cuatro módulos principales:

1. **Perfil del usuario:** se encargará de determinar el perfil al que pertenece un usuario que utiliza el sistema para conocer los aspectos fundamentales que se deben reforzar tomando en cuenta la información de las reglas de conocimiento proporcionadas al sistema.

2. **Pedagógico:** el cual contendrá la información relacionada con la descripción y uso de las herramientas del LMS, la cual será utilizada para retroalimentar al usuario.
3. **Tutoría:** se encarga de combinar la información obtenida a partir de los módulos de perfil de usuario y pedagógico para generar recomendaciones personalizadas al usuario utilizando un enfoque híbrido de tutoría.
4. **Interfaz:** será el encargado de comunicarse con el LMS y presentar por medio de un complemento integrado al LMS la información al usuario.

A continuación, se muestra un esquema de comunicación e interacción del ITS con el profesor:

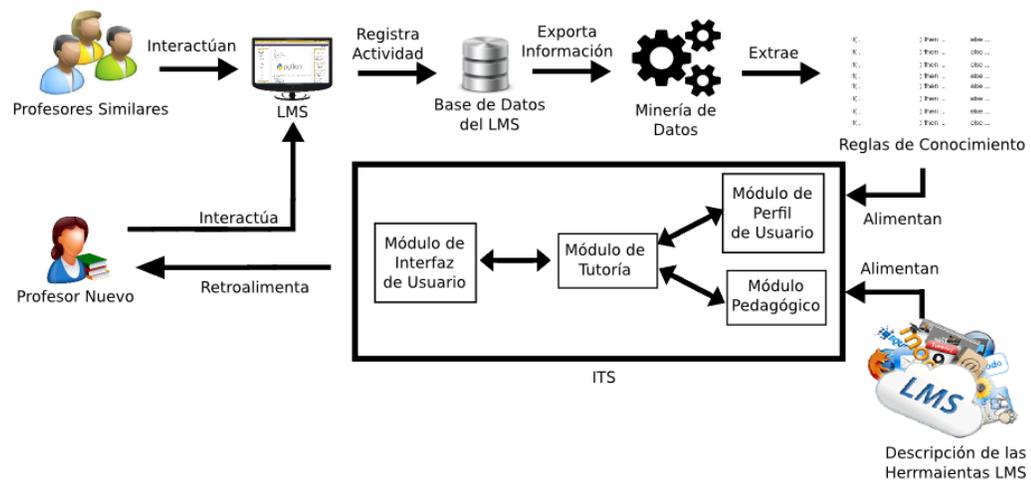


Figura 8. Esquema de comunicación e interacción entre el ITS y el profesor.

En la figura 8, se ilustra el esquema de comunicación e interacción del Tutor Inteligente con el profesor, el proceso de extracción de reglas de conocimiento y la interacción con los componentes del ITS.

El análisis de la interacción del LMS con profesores de perfiles similares permitirá la creación de perfiles iniciales que pueden ser utilizados para realizar recomendaciones a usuarios de los cuales no se tiene historial, por ejemplo, los profesores nuevos. La estrategia mencionada con anterioridad permite minimizar el impacto del arranque en frío (Son, 2016) que se da en los sistemas adaptativos cuando se carece de información del usuario.

El módulo de usuario del ITS se alimentará de los perfiles iniciales extraídos como reglas de conocimiento lo que permitirá que el ITS pueda discernir qué herramientas debe ayudar a conocer a un nuevo usuario. La descripción y los ejemplos de uso de las herramientas del LMS alimentarán el módulo pedagógico del ITS para brindar información relevante al usuario acerca de las mismas. Estas respuestas pueden ser redactadas de la manera que se considere adecuada para enriquecer la formación de los profesores e inclusive puede incluir contenido multimedia incrustado para mejorar la experiencia de interacción y la eficiencia del sistema.

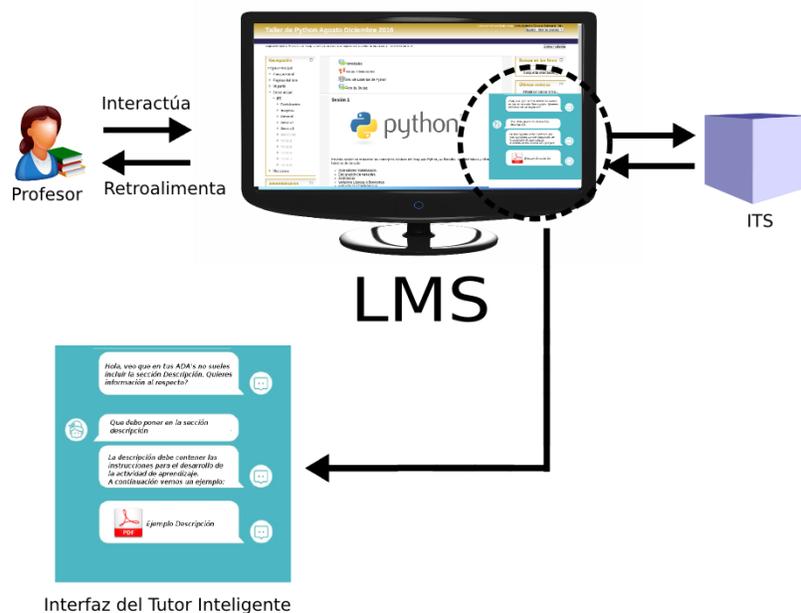


Figura 9. Esquema de interacción del ITS con el profesor.

El profesor se comunicará con el ITS por medio de una interfaz de texto incrustada en el LMS como se ilustra en la figura 9. El profesor podrá realizar preguntas en lenguaje natural que el ITS interpretará para brindar información acerca de la herramienta de interés para el profesor, enfatizando aquellas que han sido definidas como prioritarias de acuerdo a su perfil de usuario.

La interfaz además de responder con texto, podrá cargar imágenes y archivos que ayuden a profundizar el conocimiento del profesor sobre la herramienta del LMS de su interés.

### **3.2. Propuesta: MOODLE Intelligent Tutoring System (MITS)**

Para la validación del marco arquitectónico propuesto se implementó un Sistema de Tutoría Inteligente para asistir al profesor en el uso de las herramientas de MOODLE denominado MITS (MOODLE Intelligent Tutoring System).

El sistema MITS se planteó como un ITS conversacional debido a que estos sistemas han demostrado tener una buena capacidad para involucrar al usuario en el uso del sistema (Kerly, Ellis y Bull, 2008).

Adicionalmente, los sistemas conversacionales tienen la capacidad de transmitir la información de forma directa por lo que resultan útiles para consultas específicas y dudas puntuales por parte del usuario. Lo cual los convierte en un buen instrumento para interactuar con usuarios de varios niveles de conocimiento y dominio del tema como en el caso de los profesores.

Se decidió trabajar con el LMS MOODLE debido a que es de código abierto, por lo que su programación y componentes son accesibles para desarrolladores externos, además de tener una estructura sólida y desarrollada para la incorporación de componentes externos a través de complementos y plugins. Otra ventaja es el grado de penetración que tiene el entorno MOODLE en e-Learning, siendo ampliamente utilizado en todo el mundo. Esta proyección de uso del MOODLE tiene como consecuencia que las herramientas y desarrollos realizados para este entorno puedan llegar a un gran número de usuarios, lo cual permitirá una validación extensa de la propuesta por usuarios de diferentes formaciones, latitudes y formas de interactuar con los LMS.

La decisión de incrustar la interfaz de chat al entorno de MOODLE, se realizó para permitir una interacción dinámica con el usuario en el momento en el cual esté dando de alta los contenidos y permitir que el usuario pueda utilizar el sistema de tutoría sin tener que abandonar la plataforma.

El sistema está embebido dentro del entorno MOODLE como una ventana emergente ubicada en el pie de la página de la plataforma, permitiendo que el profesor pueda minimizarla y maximizarla en los momentos en que decida interactuar con el sistema. Lo anterior constituye una estrategia no invasiva para ayudar al profesor a enfocarse en sus diferentes actividades.

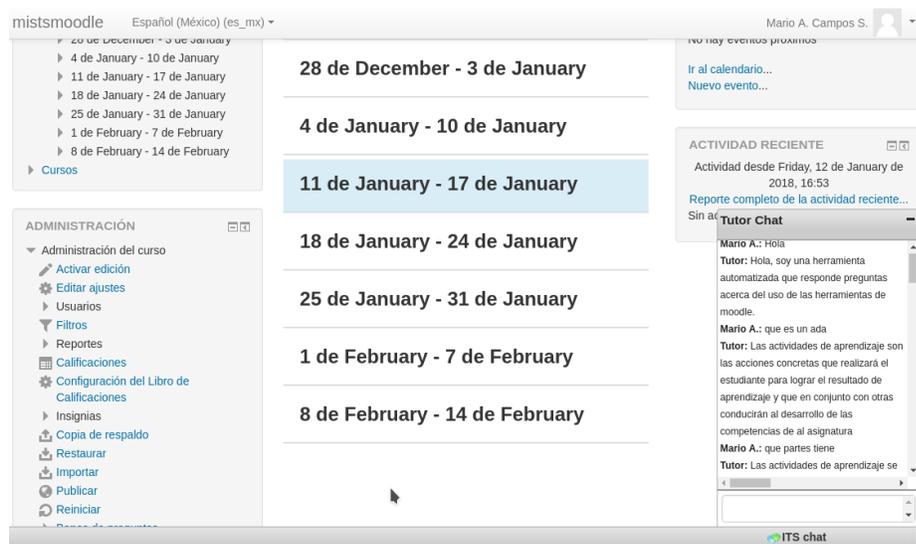


Figura 10. Interfaz del sistema MITS.

El módulo de chatbot puede dar formato a la información obtenida para incrustar elementos HTML y de esa manera enriquecer la experiencia del usuario con contenido y material interactivo.



Figura 11. Interfaz del sistema MITS con elementos multimedia incrustados.

En términos del agente conversacional se tienen dos tendencias principales: coincidencia de patrones y análisis semántico.

Los agentes conversacionales basados en reconocimiento de patrones han tenido gran éxito como lo demuestra ALICE, ganadora del premio Loebner empleando una estrategia de coincidencia de patrones mediante lenguaje de marcas de inteligencia artificial (AIML) Mikic, Burguillo, Nistal (2009). Si bien los agentes conversacionales basados en análisis semántico latente han tenido un moderado éxito como lo demuestran Kerly, Ellis y Bull (2008), se decidió implementar la técnica de reconocimiento de

patrones al ser la que mejores resultados ha reportado hasta ahora en interacciones con humanos.

La interacción con el agente conversacional se realiza a través de coincidencia de patrones buscando intenciones y entidades previamente configuradas en una base de conocimientos almacenada en archivos XML.

Esta arquitectura presenta ventajas en referencia a los chatbots implementados mediante AIML debido a la simplicidad de su modelación y facilidad de extensión hacia diferentes dominios del conocimiento permitiendo que el tutor se adapte a diferentes materias.

La resolución de las consultas efectuadas por el usuario se realiza a través de la detección de intenciones y entidades en las expresiones del usuario. La intención se refiere a la propiedad o característica que el usuario desea consultar sobre un tema específico y la entidad modela el tema de interés estudiado.

La base de conocimiento del agente se compone de archivos XML en donde se almacenan las estructuras de intenciones y entidades con sus ejemplos de consulta.

Las intenciones se modelan utilizando un archivo XML que contiene un identificador de la intención, así como ejemplos por medio de los cuales los usuarios pueden referirse a dicha intención empleando lenguaje natural.

```

<property>
  <key>partes</key>
  <synonyms>
    <synonym>DE QUE SE COMPONE</synonym>
    <synonym>QUE PARTES TIENE</synonym>
    <synonym>QUE ELEMENTOS TIENE</synonym>
    <synonym>QUE ELEMENTOS DEBE CONTENER</synonym>
    <synonym>CUALES SON SUS PARTES</synonym>
    <synonym>QUE SECCIONES TIENE</synonym>
    <synonym>COMO SE HACE</synonym>
    <synonym>ELEMENTOS</synonym>
  </synonyms>
</property>

```

Figura 12. Ejemplo de XML de intenciones.

Las entidades están modeladas como ontologías las cuales contienen propiedades con la información de los elementos que modelan. Se tiene un archivo XML por cada una de las herramientas de MOODLE acerca de las cuales el sistema puede brindar información.

```

<tool>
  <nombre>Tarea</nombre>
  <descripcion>
    El módulo de actividad tarea le permite al maestro comunicar quehaceres,
    recolectar los trabajos y proporcionar calificaciones y retroalimentación.
  </descripcion>
  <tipo_campo>
    Existe una gran variedad de herramientas externas con soporte LTI ().
    Puedes consultar algunas en
  </tipo_campo>

```

Figura 13. Ejemplo de XML de entidades.

La resolución de una consulta se realiza mediante la búsqueda de patrones de intenciones que estén contenidos en la ontología de la entidad expresada por el usuario.

Este vínculo se genera por medio del identificador de la intención el cual coincide con las propiedades modeladas en la ontología de la entidad.

En el trabajo realizado por Camacho, Zapata, Menéndez y Canto ( 2017), se analiza el uso de las herramientas de MOODLE en la Universidad Autónoma de Yucatán resultando que la actividad más empleada por los profesores es la actividad tarea. Con la finalidad de ayudar a los profesores en la construcción de las tareas en la plataforma MOODLE se decidió incluir adicionalmente información acerca del uso de la tarea y la creación de actividades de aprendizaje de acuerdo con el Modelo Educativo de Formación Integral utilizado por la UADY.

*Tabla 1.* Entidades del sistema MITS

<i>Entidades del sistema MITS</i>	
Base de Datos	Glosario
Chat	Lección
Elección	SCORM
Encuesta	Retroalimentación
Examen	Taller
Herramienta Externa	Wiki
Foro	Actividad de Aprendizaje

En la Tabla 1 se muestra las 14 entidades acerca de las cuales el sistema es capaz de brindar retroalimentación a los profesores en las cuales se incluyen las diferentes

herramientas del sistema MOODLE además de información relacionada al diseño y creación de actividades de aprendizaje de acuerdo con el MEFI.

Las intenciones que el sistema es capaz de manejar se refieren a aspectos generales de las herramientas de MOODLE así como expresiones para la consulta de temas específicos del uso de la herramienta tarea y de la elaboración de Actividades de Aprendizaje (ADA).

*Tabla 2. Intenciones del sistema MITS*

<i>Intenciones del sistema MITS</i>		
Descripción	Ejemplo	Fecha de envío
Funcionamiento	Filtros adicionales	Tipos de envío
Escenarios de Uso	Video	Tipos de ADA
Ventajas	Fecha de entrega	Planeación
Componentes	Fecha de corte	Límite de palabras
Avisar a calificador	Número de archivos	Tipos de archivo
Requerir declaratoria	Intentos reabiertos	Intentos máximos
Notificar estudiantes	Tipo de calificación	Método de calificación
Envío de grupo	Recursos	Calificación ciega
Flujograma calificador	Asignación de calificador	Disponibilidad
Comentar en línea	Hoja calificadora	Número id
Modo de grupo	Agrupamiento	Requerir botón enviar

Requerir calificación	Finalizar en	Competencias
Saludo	Categoría de calificación	Despedida
Comentarios de retroalimentación	Archivos de retroalimentación	Tamaño máximo de archivo
Calificación aprobatoria	Seguimiento de finalización	Aviso de entrega
Robot	Herramientas	

En la Tabla 2 se muestran las 50 intenciones identificadas por el sistema MITS. La información para llenar la base de conocimiento se obtuvo de la documentación de ayuda del MOODLE y del Programa Institucional de Habilitación en el MEFI.

Para encontrar las intenciones y entidades en las consultas realizadas por el usuario se aplica una medida de similitud de cadena para los elementos configurados en la base de conocimiento y por medio de una ventana deslizante verifica si el texto analizado contiene las expresiones configuradas en la base de datos tanto para las intenciones como las entidades. Si la distancia pasa un límite de similitud el módulo del agente conversacional considera que la expresión se encuentra en el texto y arma una consulta hacia la base de conocimientos para obtener la información asociada que debe contestar.

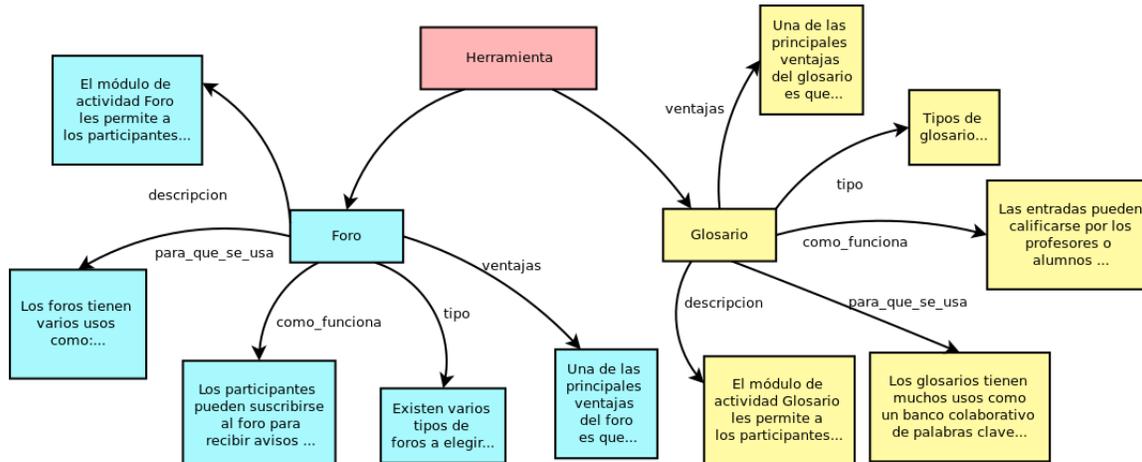


Figura 14. Ontologías de entidades del sistema MITS.

En la figura 14 se muestra un diagrama de ejemplo en donde se ilustra el modelado de la base de conocimientos de las entidades del sistema, las cuales pertenecen a la categoría de herramienta del sistema y vinculan los elementos clave definidos como intenciones del sistema con las respuestas que debe arrojar para cada una de las herramientas.

Para la medida de similitud entre las cadenas se utilizó una versión normalizada de la distancia de Levenshtein (Yujian y Bo, 2007). de la forma:

$$NED = \min \left\{ \frac{W(P_{X,Y})}{|X| + |Y|} \right\}$$

Donde  $P_{X,Y}$  es un camino de edición para transformar la palabra X en la palabra Y,  $W(P_{X,Y})$  es el peso del camino de edición  $P_{X,Y}$ ,  $|X|$  es el número de símbolos en la palabra X y  $|Y|$  es el número de símbolos en la palabra Y. Se escogió esta distancia debido a que ha sido reportada como una buena medida de similitud entre cadenas siendo ampliamente

utilizada en reconocimiento de patrones de texto, además de proporcionar tolerancia a errores tipográficos y a elementos semánticos complejos.

Adicionalmente se decidió dotar al sistema MITS con un manejo de contextos y memoria para añadir naturalidad a la comunicación y mejorar la experiencia del usuario. Lo anterior se realiza mediante un algoritmo propuesto el cual verifica el tema del cual se está hablando con el usuario desde la perspectiva intención y entidad para realizar la consulta respectiva y almacenar en la base de datos el sistema el último contexto del usuario.

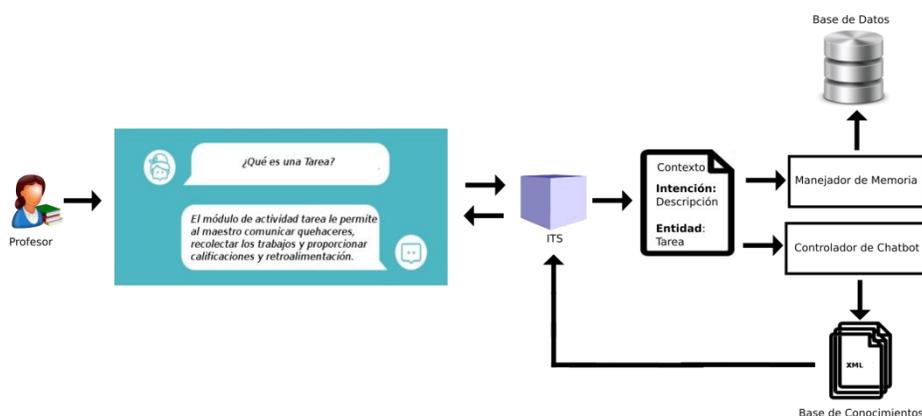


Figura 15. Construcción del contexto del usuario.

La estrategia de manejo de memoria considera la última intención y entidad acerca de la cual el usuario realizó una consulta cuya combinación genera un contexto a partir del cual el tutor puede consultar información en caso de que el usuario no exprese explícitamente alguna de ellas en la siguiente consulta. En caso de que el usuario no especifique una intención o una entidad, se considera que ésta implícita en el último

contexto y se obtiene la información faltante del contexto almacenado en la base de datos para realizar la consulta a la base de conocimientos.

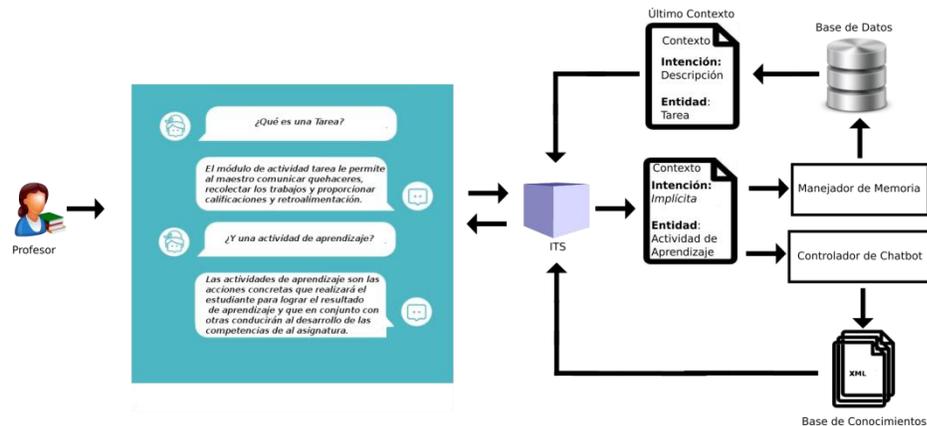


Figura 16. Construcción del contexto del usuario con información implícita.

En la figura 16 se muestra el tratamiento del sistema para la construcción de un contexto con información implícita. Una vez que se realiza la combinación del contexto actual y del último contexto, el sistema almacena la nueva información mediante el manejador de memoria y recupera la información de la base de conocimientos por medio del controlador del chatbot.

```

if $I_USUARIO exists in $USER_QUERYy && $E_USUARIO exists in $USER_QUERY:
    $contexto = [$I_USUARIO, $E_USUARIO]
else
    $previous_context = [$I_PREVIOUS, $E_PREVIOUS]
    $contexto = [$I_USUARIO OR $I_PREVIOUS, $E_USUARIO OR $E_PREVIOUS]

$respuesta = kb($contexto)
updateContext($contexto)

```

Figura 17.Pseudocódigo del algoritmo de extracción de contextos.

En la figura 17 se muestra el pseudocódigo del algoritmo de la construcción de contextos en donde \$I\_USUARIO representa la intención explícita del usuario, \$E\_USUARIO es la entidad explícita del usuario, \$I\_PREVIOUS es la intención anterior del usuario almacenada en la base de datos del contexto, \$E\_PREVIOUS es la entidad anterior almacenada en la base de datos de contextos, kb(\$contexto) es una función que realiza la consulta a la base de conocimientos del sistema utilizando el contexto pasado como parámetro y updateContext(\$contexto) es una función que almacena en la base de datos el nuevo contexto generado para el usuario.

Lo anterior queda expresado en el diagrama de flujo de la figura 18:

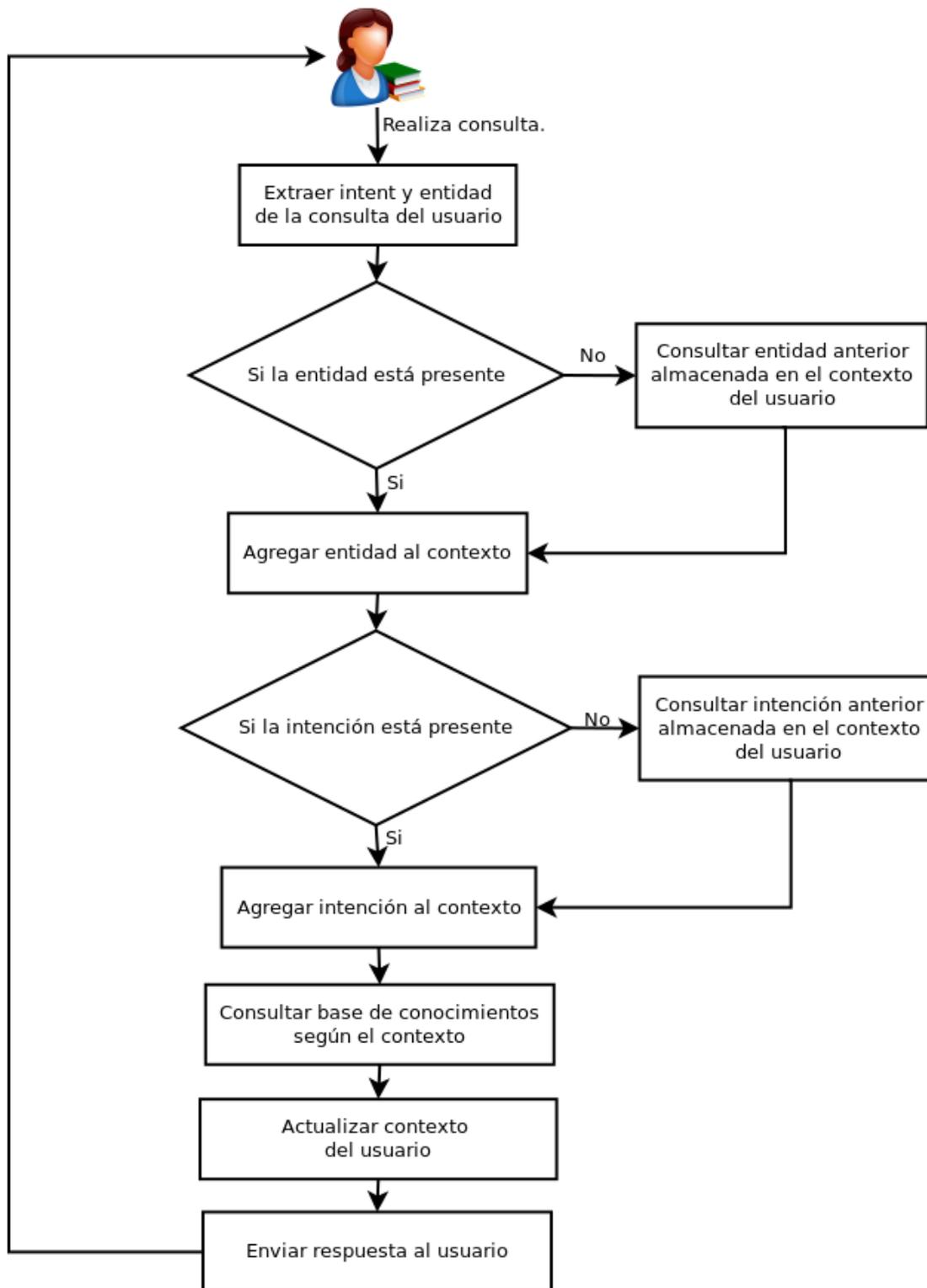


Figura 18. Diagrama de flujo del algoritmo de extracción de contextos.

En términos de la estrategia de tutoría se decidió utilizar un tutor inteligente híbrido que incorpore las técnicas de secuencia de currículo y tutoría basada en restricciones. Las motivaciones para implementar estas estrategias de tutoría obedecen al hecho de que la secuencia de currículo permite personalizar el aprendizaje de acuerdo con las preferencias del usuario lo cual puede resultar de gran utilidad debido a los diferentes dominios de conocimiento, asignaturas impartidas y experiencia de los docentes. La estrategia de tutoría basada en restricciones ha sido reportada por Weerasinghe et al. (2009) como un método útil de aproximarse a tutoría en dominios del conocimiento débilmente definidos.

La estrategia de tutoría de secuencia de currículo se implementa utilizando el funcionamiento natural del chatbot permitiendo que el usuario interactúe y pregunte acerca de los temas de su interés, lo que permite que el usuario personalice su entrenamiento de acuerdo con sus necesidades.

Para la implementación del tutor basado en restricciones se construye un perfil del usuario de acuerdo con el uso de las herramientas que el profesor tiene en la plataforma MOODLE. Este perfil se compara con los perfiles iniciales retroalimentados al sistema por medio del módulo de perfil del usuario. Se considera como prioritaria aquella herramienta que presenta la mayor distancia de acuerdo con el perfil del profesor en comparación con el perfil inicial en el cual fue clasificado. Cada profesor es representado por un vector de uso de las herramientas en donde cada una es calificada con una escala de uso del 1 al 5, donde 1 indica un uso muy bajo de la herramienta y 5 un uso muy alto de la misma. Esta representación tiene la ventaja de ser escalable en cuanto el número de

herramientas que se desean utilizar para modelar el perfil del usuario. Para clasificar el uso de una herramienta por parte del profesor el sistema MITS se alimenta con los vectores de uso obtenidos del análisis de la interacción de los profesores con MOODLE, donde cada vector consta de entradas del 1 al 5 que indican el uso de cada herramienta de acuerdo con la posición.

Adicionalmente, se puede proporcionar un vector que indica los diferentes niveles de uso. Lo anterior permite que los valores de la escala puedan ser definidos al momento de implementar el sistema, e inclusive manejar diferentes valores de uso de acuerdo con cada herramienta del sistema. Este mecanismo se puede utilizar para definir diferentes escalas y criterios para el perfil del usuario.

La información del uso de la herramienta por parte del profesor se extrae del registro de sus acciones en el sistema MOODLE y se obtiene sumando el número de interacciones de sus últimos cursos, siendo el número de cursos a considerar configurable. El sistema verifica para cada uno de los cursos considerados el uso de las herramientas y excluye valores atípicos que pudieran sesgar el perfil. Si el valor de uso de alguna herramienta excede dos desviaciones estándar del uso normal del profesor el valor no es tomado en cuenta para la elaboración del perfil.

Una vez que se decide el nivel de uso de cada herramienta para el profesor se genera un vector de características formado por los diferentes niveles de uso para cada herramienta. Para definir el perfil se verifica la distancia entre el vector de uso del profesor y los perfiles proporcionados al sistema.

Para medir la distancia entre los vectores de perfil y el de profesor se tomaron en cuenta dos medidas de similitud: la diferencia promedio normalizada y la distancia de coseno suave.

$$\text{soft\_cosine}(a, b) = \frac{\sum_{i,j}^N s_{i,j} a_i b_j}{\sqrt{\sum_{i,j}^N s_{i,j} a_i a_j} \sqrt{\sum_{i,j}^N s_{i,j} b_i b_j}}$$

Donde  $s_{i,j}$  indica la similitud entre la característica  $i$  y la característica  $j$ ,  $a_i$  el  $i$ ésimo elemento del vector  $a$  y  $b_i$  el  $i$ ésimo elemento del vector  $b$ . Se eligió esta medida de comparación debido a los buenos resultados reportados por Sidorov, Gelbukh, Gómex-Adorno y Pinto (2014) para la comparación de características específicas dentro de espacios vectoriales, por lo cual añade versatilidad a la hora de realizar las comparaciones, pudiendo definir los grados de similitud por parte del usuario.

De igual manera, verificamos la similitud entre vectores utilizando el valor promedio de las diferencias normalizadas entre las diferencias de los elementos de los vectores.

$$\text{NAD} = \frac{\sum_1^N 1 - \frac{|a_i - b_i|}{S}}{N}$$

Donde  $S$  indica la escala de los valores máximos del vector,  $a_i$  el  $i$ ésimo elemento del vector  $a$  y  $b_i$  el  $i$ ésimo elemento del vector  $b$ . Esta métrica fue seleccionada debido a su bajo costo computacional y por ser una alternativa viable para comparaciones de perfil simples.

Una vez que el perfil del usuario se ha decidido, se elige que herramienta retroalimentar al usuario calculando la mayor diferencia entre las herramientas del perfil y las herramientas del vector del usuario.

### **3.3. Arquitectura del sistema**

El sistema MITS se desarrolló utilizando una arquitectura modular con el objetivo que sus diferentes elementos puedan ser reutilizados o remplazados con lo cual se agrega versatilidad para probar diferentes técnicas en sus componentes. Lo anterior permite explorar diferentes variantes de tutoría, agente conversacional o dominio de conocimiento implementando el componente deseado. Dicho sistema contiene cuatro módulos principales (fig. 19).

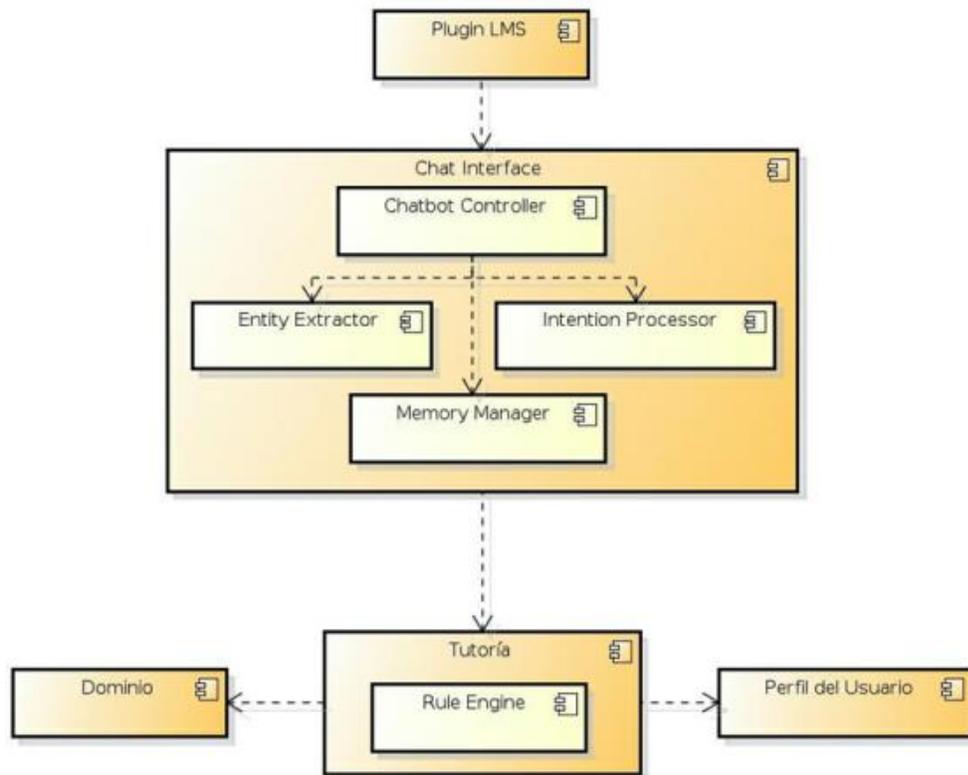


Figura 19.. Componentes del sistema MITS.

En la figura 19, se observan la forma como interactúan los cuatro módulos de la arquitectura propuesta, los cuales se describen a continuación:

- **Módulo de interfaz del chatbot.** Es el encargado de procesar las entradas de texto, formatear las salidas y entregar la información al usuario final. Se implementó utilizando un agente conversacional de coincidencia de patrones que realiza la búsqueda de términos clave a través de una comparación difusa de cadenas (Weerasinghe et al., 2009). Este método se seleccionó por tratarse de una herramienta robusta a variaciones gramaticales y a errores ortográficos comunes en este tipo de sistemas.

- **Módulo de tutoría.** Se implementó un sistema de tutoría híbrido que permite secuencia de currículo mediante búsquedas específicas por parte del usuario y tutoría basada en restricciones (Weerasinghe et al., 2009), esto significa, que verifica si el usuario cumple un conjunto de reglas. Para efectos de esta propuesta, la información de uso de las herramientas de MOODLE por parte del profesor, y se asigna un valor a cada regla. En caso de que el valor no llegue a un límite aceptable se rompe una restricción y se determina que se requiere proporcionar retroalimentación al usuario sobre el tema. En la situación de un usuario nuevo, se verifican las reglas de conocimiento previamente extraídas que modelan el comportamiento de perfiles similares. Cuando el profesor no haya utilizado las herramientas lo suficientemente para generar una regla, el tutor inteligente decidirá proporcionar información acerca de las que no alcanzaron el valor mínimo. Finalmente, cuando el profesor ya ha utilizado el sistema y no rompe las reglas se determina que herramientas utiliza menos para brindar información relativa a ellas.
- **Módulo de perfil de usuario.** Se encarga de extraer la información relativa al usuario y el uso que realiza con las herramientas del software MOODLE para proporcionarle la información al módulo de tutoría. La extracción de los perfiles se realiza conectándose directamente a la base de datos de MOODLE y verificando el uso de las herramientas por parte del profesor.
- **Módulo de acceso al dominio.** Realiza la consulta de la información solicitada por el usuario en la base de conocimiento y devuelve las propiedades solicitadas.

La base de conocimiento en el sistema es modelada por medio de ontologías de las herramientas del software MOODLE, dichas ontologías son construidas con las descripciones que el sistema proporciona por medio de las herramientas.

El módulo de interfaz de chat tiene un componente principal el cual es el chatbot controller, el cual se encarga de comunicar la información a los diferentes componentes que conforman el módulo. El componente EntityExtractor consulta las entidades del sistema y realiza la extracción de las mismas de la consulta del usuario. El componente IntentionProcessor se encarga de extraer las intenciones de la consulta del usuario y el componente MemoryManager se encarga de la actualización de los contextos del usuario.

El módulo de tutoría tiene un componente llamado RuleEngine que se encarga de verificar que herramienta se debe de retroalimentar de acuerdo con el perfil del usuario y a los perfiles alimentados inicialmente.

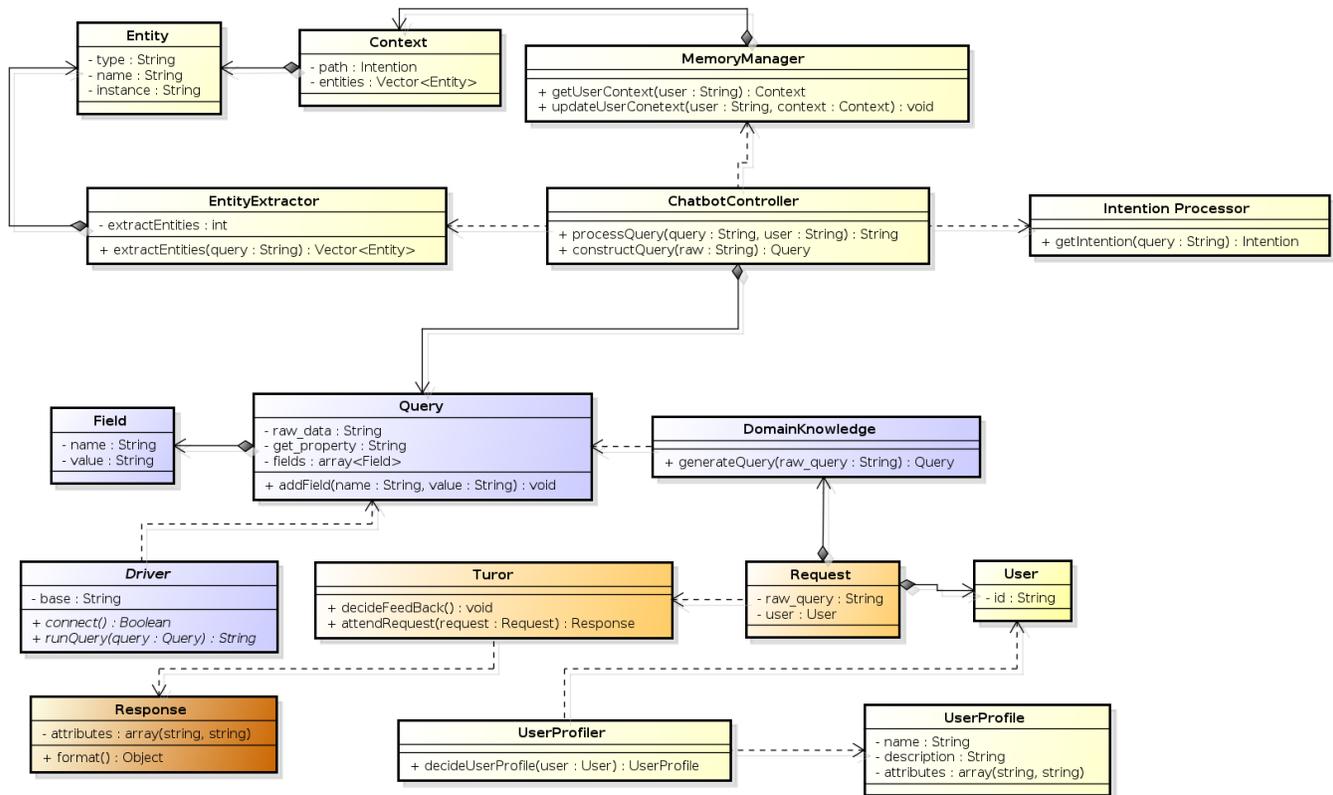


Figura 20. Diagrama de clases del sistema MITS.

En la figura 20 se observa el diagrama de clases con el cual se implementan los diferentes módulos y componentes del sistema MITS. Las clases se encuentran desacopladas por lo cual resulta sencillo implementar partes específicas del sistema para experimentar con diferentes estrategias de tutoría, reconocimiento de patrones, agente conversacional o perfilador de usuario. Lo anterior presenta una característica interesante puesto que abre la puerta a comparación de diferentes maneras de implementar MITS.

El desarrollo del prototipo se realizó utilizando técnicas de programación orientada a objetos (POO) en la cual se modelaron en primera instancia los elementos clave de la arquitectura en una estructura llamada GPITS (General Purpose Intelligent

Tutoring System). Dicha estructura contiene las clases *Entity*, *Context*, *MemoryManager*, *ChatBotController*, *EntityExtractor*, *Query*, *Field*, *Driver*, *KnowledgeDomain*, *Request*, *Response*, *Tutor*, *User*, *UserProfile* y *UserProfiler* las cuales son versiones abstractas que definen los métodos y atributos esenciales para la creación de un ITS de acuerdo con la arquitectura general propuesta en este trabajo.

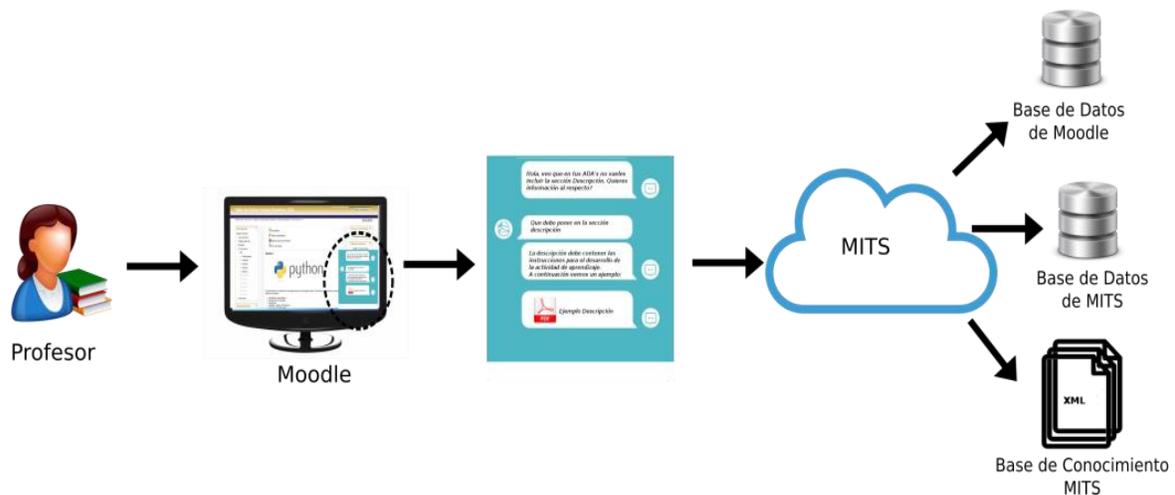
El sistema MITS se implementó creando instancias específicas para las clases del core del sistema GPITS, siendo las más relevantes el *MITSMemoryManager* que utiliza la estrategia de contextos antes citada, *MITSEntityExtractor* la cual se encarga de la extracción de las entidades utilizando un *XMLDriver* que realiza la búsqueda difusa de cadenas de acuerdo al algoritmo normalizado de Levenshtein y la técnica de ventana deslizante, *MITSController* que se encarga del flujo de información entre los diferentes módulos del sistema, *MITSUserProfiler* que se encarga de extraer los perfiles de usuario de acuerdo al uso de las herramientas por parte del profesor y *MITSTutor* que implementa las estrategias de tutoría mencionadas a lo largo de este trabajo.

### **3.4. Implementación técnica de la propuesta**

Para la implementación técnica de la propuesta se consideró un desarrollo modular partiendo de una base genérica y abstracta de las clases con la intención de proveer una estructura general para desarrollar ITS que puedan adaptar diferentes tipos de conocimiento y formas de expresarlo.

MITS se implementó como un servicio web desarrollado en PHP que interactúa con las bases de datos de MOODLE, la base de datos de contextos y la base de conocimientos del sistema implementada a través de archivos XML.

La propuesta se implementó utilizando como LMS el sistema MOODLE en la versión 3.1.6, MySQL 5.5.58 y PHP 5.5.9.



*Figura 21. Implementación del sistema MITS.*

En la figura 21 se ilustra como el sistema interactúa con el usuario por medio de un chat incrustado en el MOODLE utilizando el plugin mitschat, el cual fue desarrollado tomando como base el plugin VMchat, que se conforma por una serie de componentes implementados en PHP y Javascript. Las bibliotecas que se comunican directamente con el sistema MOODLE se implementan en tecnología PHP y la interfaz de chat en javascript.

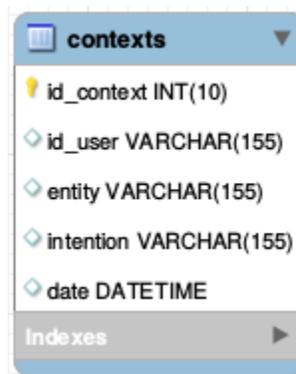
Para realizar la consulta al sistema MITS el módulo de la biblioteca en PHP define una serie de variables en el módulo de Javascript permitiendo el paso de la información del usuario y perfil.

Posteriormente la biblioteca de Javascript realiza una llamada AJAX a través de la biblioteca JQuery al servicio web de MITS, el cual utiliza la información obtenida de las bibliotecas de MOODLE para realizar el procesamiento de la entrada del usuario pasando por diferentes módulos.

En primera instancia se pasa la información al manejador de memoria que realiza una consulta al controlador del chatbot quien se comunica con la base de conocimiento de MITS para verificar las entidades e intenciones proporcionadas por el usuario.

Adicionalmente el manejador de memoria consulta contexto almacenado en su base de datos para el usuario.

La base de datos del manejador de memoria consiste en dos tablas: context e interactions.



The image shows a screenshot of a database table structure for a table named 'contexts'. The table has five columns: 'id\_context' (INT(10)), 'id\_user' (VARCHAR(155)), 'entity' (VARCHAR(155)), 'intention' (VARCHAR(155)), and 'date' (DATETIME). The 'id\_context' column is marked as the primary key with a yellow lightning bolt icon. Below the columns, there is a section for 'Indices' with a right-pointing arrow.

Column Name	Data Type	Primary Key
id_context	INT(10)	Yes
id_user	VARCHAR(155)	No
entity	VARCHAR(155)	No
intention	VARCHAR(155)	No
date	DATETIME	No

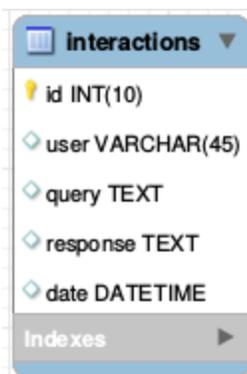
Figura 22. Tabla de contextos del sistema MITS.

La tabla contexts almacena información del contexto. Los registros de ésta tabla se componen de un campo entero sin signo, implementada como clave primaria de las entradas, id\_user implementado como una variable VARCHAR(155) que almacena el id del usuario en el sistema, la variable intention de tipo VARCHAR(155) almacena el

identificador de la última intención expresada por el usuario, la variable intention de tipo VARCHAR(155) almacena la última entidad almacenada por el usuario y la variable date de tipo DATETIME que almacena la fecha y hora de la última actualización del contexto.

La información de las entidades e intenciones, junto con el contexto almacenado pasan al sistema MITS que construye el nuevo contexto y mediante un Driver XML implementado en PHP genera una consulta para obtener la información a retroalimentar al usuario.

Una vez obtenida la información necesaria para retroalimentar al usuario la interacción se almacena en la tabla interactions del sistema MITS.



The image shows a screenshot of a database table definition for 'Interactions'. The table has the following columns:

Column Name	Data Type
id	INT(10)
user	VARCHAR(45)
query	TEXT
response	TEXT
date	DATETIME

At the bottom of the screenshot, there is a section labeled 'Indexes' with a right-pointing arrow, indicating that the table has one or more indexes.

Figura 23. Tabla de interacciones del sistema MITS.

La tabla interactions almacena todas las interacciones ocurridas dentro del sistema. Los registros de esta tabla se componen de un campo entero sin signo, implementado como clave primaria de las entradas llamado id, el campo user implementado como una variable VARCHAR(45) que almacena el id del usuario que realiza la interacción, la variable query de tipo TEXT almacena la entrada del usuario, la

variable response de tipo TEXT almacena la respuesta del sistema y la variable date de tipo DATETIME que almacena la fecha y hora de la interacción.

El sistema MITS tiene dos funciones principales: la respuesta a una consulta del usuario y la recomendación de acuerdo al perfil.

*Tabla 3. Secuencia de ejecución para responder preguntas del sistema MITS*

<i>Respuesta de preguntas del sistema MITS</i>	
<b>Paso</b>	<b>Acción</b>
1	Consulta a la base de conocimientos del sistema.
2	Procesamiento de la consulta del usuario.
3	Consulta a la base de datos de contextos.
4	Construcción del contexto actual.
5	Actualización de la base de datos de contextos.
6	Consulta a la base de conocimientos del sistema.
7	Formateo de la respuesta para el usuario.

En la tabla 3 se muestra la secuencia de ejecución mediante la cual el sistema realiza la respuesta a las consultas específicas del usuario. En primera instancia se consulta la base de datos del sistema para extraer las intenciones y entidades configuradas en la base de conocimiento. Esta información se utiliza para procesar la consulta del usuario y mediante el algoritmo de Levenshtein normalizado (Yujian y Bo, 2007) se verifica cuáles de las intenciones y entidades configuradas en el sistema se encuentran en

la consulta del usuario. El siguiente paso de la secuencia es consultar a la base de contextos para verificar que intención y entidad tiene en memoria el sistema. Utilizando la información proporcionada por el usuario y el contexto anterior se construye el nuevo contexto y se actualiza en la base de datos. A continuación, se consulta a la base de conocimiento para encontrar la respuesta que se debe proporcionar al usuario y se formatea para incluir los elementos incrustados. Esta información se envía de vuelta al plugin mitschat quien la presenta al usuario por medio del chatbot.

Cuando el sistema emite una recomendación utilizando la información del usuario el sistema sigue una secuencia de ejecución diferente.

*Tabla 4. Secuencia de ejecución para emitir recomendaciones del sistema MITS*

<i>Recomendaciones del sistema MITS</i>	
<b>Paso</b>	<b>Acción</b>
1	Consulta a la base de datos de MOODLE.
2	Extracción de perfil del usuario.
3	Definición de herramienta a retroalimentar.
4	Consulta a la base de conocimientos del sistema.
5	Formateo de la respuesta para el usuario.
6	Recomendación de recursos al usuario de acuerdo al perfil.

En la tabla 4 se muestra la secuencia de ejecución del sistema para emitir recomendaciones personalizadas para el usuario utilizando su perfil. En primera instancia

se verifica la actividad del profesor en el sistema MOODLE. Con esa información se construye el vector de características del usuario descrito por el uso de las diferentes herramientas. Utilizando el perfil del usuario y los perfiles iniciales cargados en el sistema se determina la herramienta que se debe retroalimentar al usuario utilizando la mayor diferencia entre el perfil del usuario y los perfiles iniciales del sistema. A continuación, el sistema realiza una consulta a la base de conocimientos para dar información acerca de la herramienta y efectúa el formateo de dicha información para incrustar los elementos adicionales y emitir la recomendación al usuario.

El sistema se implementó en la tecnología Google Cloud, estando accesible para los usuarios a través de internet.

## **Capítulo 4. Experimentación y resultados.**

Este capítulo tiene como propósito presentar la validación de los resultados obtenidos de la investigación mediante experimentos automáticos y con usuarios reales.

Se inicia evaluando las métricas de similitud de perfiles mediante pruebas automatizadas y posteriormente se evalúa la usabilidad del sistema MITS por parte de los usuarios usando la Escala de Usabilidad del Sistema (SUS) .

### **4.1 Pruebas automatizadas.**

Para personalizar las recomendaciones hacia los usuarios el sistema necesita información generada por el profesor mediante su interacción con MOODLE, sin embargo, en el entorno en el que se realiza la presente investigación se carece de un historial de uso de las herramientas por parte de los profesores.

Con el fin de probar el desempeño de MITS como tutor inteligente capaz de personalizar los contenidos mostrados al profesor de acuerdo con su perfil, se generó un caso de uso automatizado en donde se introducen perfiles iniciales y se verifica la clasificación del perfil del usuario según las métricas de diferencia promedio normalizada y similitud de coseno suave descritas previamente.

Para la experimentación de este caso se cargaron previamente 3 perfiles de usuario predefinidos con 10 herramientas del sistema:

*Tabla 5. Perfiles predefinidos de usuario del sistema MITS*

<i>Perfiles predefinidos del sistema MITS</i>	
<b>Perfil</b>	<b>Datos</b>
1	[5,5,5,1,1,1,1,1,1,1]
2	[1,1,1,5,5,5,5,1,1,1]
3	[1,1,1,1,1,1,1,5,5,5]

En la tabla 5 se muestran los perfiles de uso de las herramientas de parte del sistema, los cuales fueron seleccionados para ser claramente diferenciables y facilitar su clasificación por parte de un humano experto poniendo énfasis en un uso muy alto de ciertas herramientas y muy bajo en las demás. El primer perfil muestra a un profesor que realiza un uso muy alto de las tres primeras herramientas mientras que tiene un uso muy bajo para las demás. El segundo perfil se caracteriza por hacer un uso muy elevado de las herramientas 4,5,6 y 7 mientras que realiza un uso muy bajo de las demás. El tercer perfil realiza un uso muy alto de las herramientas 8,9 y 10 mientras que utiliza muy poco las demás.

Para probar la eficiencia de la clasificación del módulo de perfil del usuario se generaron 30 ejemplos de manera automática los cuales fueron clasificados manualmente en uno de los tres perfiles predefinidos con la asesoría de un experto.

A continuación, se realizó la clasificación automática por medio del sistema MITS utilizando las medidas de diferencia promedio normalizada y la distancia de coseno suave.

*Tabla 6.* Resultados de pruebas automatizadas

<i>Pruebas automatizadas</i>		
<b>Medida</b>	<b>Aciertos</b>	<b>Porcentaje</b>
Diferencia promedio normalizada	27	90.00 %
Similitud de coseno suave	28	93.33 %

De acuerdo a los resultados obtenidos en las pruebas automatizadas (Tabla 6) el sistema mostró un buen desempeño en la clasificación de los perfiles de los usuarios siendo éstos superiores al 90%. Un resultado destacable de las pruebas automatizadas es que la similitud de coseno suave demostró un desempeño ligeramente mejor que la diferencia promedio normalizada, sin embargo, ésta diferencia no es significativa para la clasificación utilizando el vector de características generado por el sistema.

#### **4.2 Diseño del experimento.**

El entorno de la experimentación es la Universidad Autónoma de Yucatán, donde se realizaron dos pruebas de desempeño del sistema MITS. La primera, se realizó con un grupo de profesores de la Facultad de Educación quienes cuentan con una amplia experiencia utilizando diversos LMS en cursos de licenciatura y posgrado. Adicionalmente, se destaca que están habilitados para dar clases en los programas educativos alineados al MEFI en la UADY.

La segunda prueba, fue con el personal del Departamento de Innovación e Investigación Educativa de la institución, quienes entre sus funciones son los encargados de implementar el uso del LMS UADY virtual (basado en el software MOODLE) entre los profesores de la UADY. Dicho sistema actualmente cuenta con más de 14,000 cuentas de usuario (entre profesores y alumnos) y se está utilizando como una herramienta de apoyo a las clases presenciales en los niveles de licenciatura y posgrado en esta institución.

Para medir la facilidad de uso del sistema MITS se utilizó la métrica SUS (System Usability Scale) la cual consta de un cuestionario de 10 reactivos (Tabla 7) con cinco respuestas cada uno, que van desde “totalmente de acuerdo” hasta “totalmente en desacuerdo” (Brook, 1996). Se utilizó esta prueba dado que es sencilla de administrar a los participantes, puede ser usada en pequeños grupos y puede diferenciar entre un sistema usable y uno que no lo es (usability.gov, 2017 ). Cabe mencionar que de las 10 preguntas propuestas, los números impares son clasificados como positivos y los números pares considerados como negativos en el ámbito de la facilidad de uso del sistema (Bangor et. al, 2009). Si bien el rango que arrojan los resultados de la prueba va del 1 a 100, éste no debe interpretarse como un “porcentaje de facilidad de uso”, sino que debe “normalizar” los resultados y generar un rango de percentiles. Sin embargo, el hecho de tener una calificación mayor a 70 en la prueba categoriza a la facilidad de uso del sistema como una aceptable, mayor a 85 como excelente e igual a 100 como “la mejor imaginable” (Carbonero et. al, 2009).

De acuerdo con (Tullis y Stetson, 2006), con un pequeño grupo muestra (de 8 a 12 participantes) se pueden obtener resultados confiables de la percepción de facilidad de uso de un sistema, por lo tanto, se eligió una muestra pequeña de participantes que fueran potenciales usuarios del sistema en las dos pruebas de desempeño.

La experimentación se llevó a cabo con 11 participantes de los cuales 5 pertenecen a la Facultad de Educación de la UADY y 6 al Departamento de Investigación e Innovación Educativa de la UADY. Del total de participantes 1 reportó tener licenciatura como grado máximo de estudios, 7 maestría y 3 doctorado. Con respecto al área de formación 3 de los usuarios pertenecían al área de ingeniería y tecnología, 1 al área de ciencias sociales y administrativas y 7 al área de educación, humanidades y arte. En cuanto a la experiencia docente de los participantes 4 de los usuarios indicaron tener una experiencia menor a 5 años, 5 de los participantes cuentan con una experiencia docente y 2 entre 10 y 15 años de experiencia. En la experiencia interactuando con sistemas de gestión del aprendizaje 3 cuentan con una experiencia menor a los 5 años, 6 tienen una experiencia entre 5 y 10 años y 2 entre 10 y 15 años.

Como se comentó anteriormente, la experimentación se llevó a cabo en dos momentos. Primero en las instalaciones de la Facultad de Educación y posteriormente, en el edificio del Departamento de Investigación e Innovación Educativa de la UADY. En ambos casos, se consultó por medio de equipos locales un servidor de MOODLE con el plugin mitschat instalado. Dicho servidor se instaló utilizando la nube de Google, para ser accesible desde internet. La instancia del servicio del sistema MITS se configuró en el mismo servidor que corría la plataforma MOODLE.

El experimento se elaboró pidiendo a los participantes que elaboraran una Actividad de Aprendizaje de su preferencia en el sistema MOODLE después de interactuar con el sistema MITS. La selección de esta herramienta, se debe a que en el trabajo Camacho et al. (2017), se encontró que el desarrollo de las Actividades de Aprendizaje es lo que más registra actividad en la UADY y ocupa un 80 % de lo que se almacena en UADY.

Se realizó una breve inducción al funcionamiento del sistema MITS indicando a los profesores la manera de interactuar con el mismo. Los usuarios realizaron las preguntas que consideraron pertinentes para el desarrollo de la actividad de aprendizaje y posteriormente realizaron la creación y configuración de su actividad en el sistema. La prueba tuvo una duración de 30 minutos. A continuación, se presentan las instrucciones del experimento como fueron presentadas a los usuarios:

1.- Iniciar sesión en el MOODLE <http://35.227.27.90/moodle> usando el usuario usuario2 y la contraseña Usu@rio2 e ingresar al Curso de prueba 2 MITS.

2.- Crear una Actividad de Aprendizaje proporcionada por el usuario (requisito para la experimentación) a través del uso del Tutor MITS.

3.- Conteste el siguiente cuestionario relacionado a su experiencia al interactuar con el sistema MITS.

*Tabla 7. Escala de usabilidad del sistema SUS*

<i>SUS</i>
<i>Creo que me gustaría utilizar este sistema frecuentemente.</i>

<i>Encuentro este sistema innecesariamente complejo.</i>
<i>Pienso que el sistema es fácil de usar.</i>
<i>Creo que necesitaría soporte técnico para hacer uso del sistema.</i>
<i>Encuentro las diversas funciones del sistema bastante bien integradas.</i>
<i>He encontrado demasiada inconsistencia en este sistema.</i>
<i>Creo que la mayoría de la gente aprendería a hacer uso del sistema rápidamente.</i>
<i>He encontrado el sistema bastante incómodo de usar.</i>
<i>Me he sentido muy seguro haciendo uso del sistema.</i>
<i>Necesitaría aprender muchas cosas antes de manejar el sistema.</i>

### 4.3 Resultados.

Tras una consulta de la actividad de los usuarios con el sistema, se realizaron 252 interacciones promediando 19.38 interacciones por cada uno.

De las 252 interacciones realizadas con el sistema 112 se encontraban dentro de la base de conocimientos del tutor MITS, de las cuales contestó correctamente 92 interacciones obteniendo un 82.14 % de respuestas correctas para las preguntas dentro de su dominio de conocimiento.

Entre las preguntas que los usuarios realizaron fuera del dominio de conocimiento del sistema se encontraron cuatro categorías diferentes:

1.- **Creación y diseño:** Los usuarios preguntaron acerca de las formas de crear elementos y actividades dentro del sistema, así como la manera adecuada de diseñarlas. Un ejemplo de pregunta de esta categoría es: “¿Cómo diseño una tarea?”.

2.- **Recursos y opciones:** Los usuarios preguntaron acerca de recursos que pudiera contener el sistema relacionados a la creación de los elementos en la plataforma

así opciones específicas de las herramientas. También solicitaron asesoría para la creación de sus propios recursos de aprendizaje. Un ejemplo de pregunta de esta categoría es: “Pasos para hacer un recurso”.

3.- **Edición de actividades:** Los usuarios preguntaron acerca de las formas para realizar la edición de actividades del sistema. Una pregunta de ejemplo es “¿Qué es activar edición?”.

4.- **Opciones varias:** Los usuarios realizaron preguntas de variada naturaleza incluyendo herramientas no consideradas en el dominio de conocimiento del sistema como rúbrica y libro de calificaciones. Además, los usuarios hicieron consultas para verificar si el sistema podía realizar análisis de sus cursos. Como ejemplo de una pregunta de esta categoría podemos citar “¿Qué hay programado en el curso de prueba MITS?”.

*Tabla 8. Interacciones no incluidas en el dominio de conocimiento del sistema MITS*

<i>Categoría</i>	<i>Número de preguntas</i>
Opciones varias	51
Creación y diseño	47
Recursos y opciones	27
Edición de actividades	15

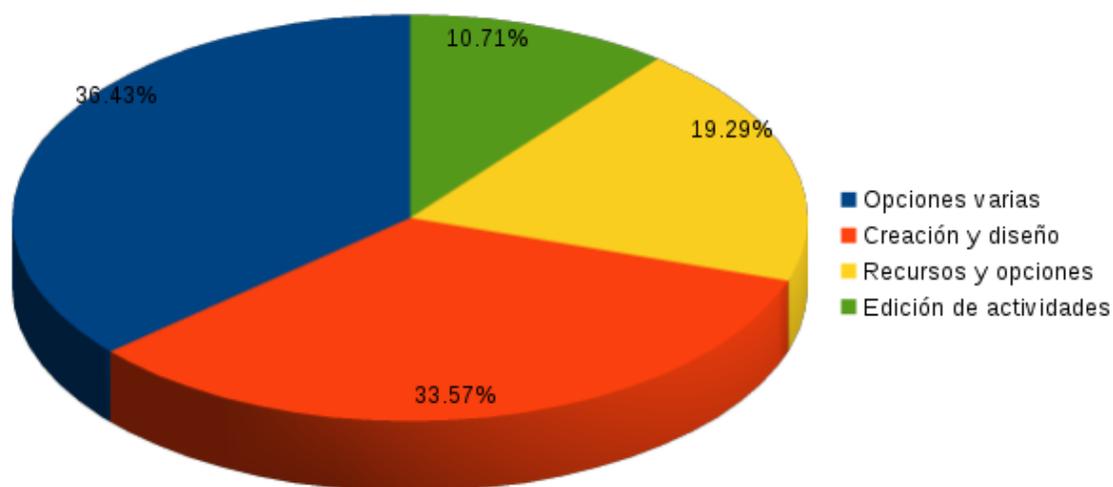


Figura 24. Gráfica de interacciones no incluidas en el dominio de conocimiento del sistema MITS.

A partir de los resultados presentados en la figura 24 se observa que la creación y diseño de actividades es un tema de interés común entre los profesores.

En cuanto a la escala SUS se alcanzó validez experimental debido a que se desarrolló el experimento utilizando el número de usuarios adecuado.

Tabla 9. Resultados de la prueba SUS del sistema MITS

<i>Calificaciones SUS</i>
57.5
67.6
87.5
90



Figura 25. Gráfica de resultados de la escala SUS del sistema MITS.

En la figura 25 se muestra una gráfica con las calificaciones obtenidas en la escala de usabilidad SUS. 82% de los participantes obtuvieron resultados mayores a 65 en la escala indicando una usabilidad aceptable, sin embargo, el 18% restante obtuvo resultados por debajo de 65, lo cual influyó para que el sistema no fuera considerado con una usabilidad excelente.

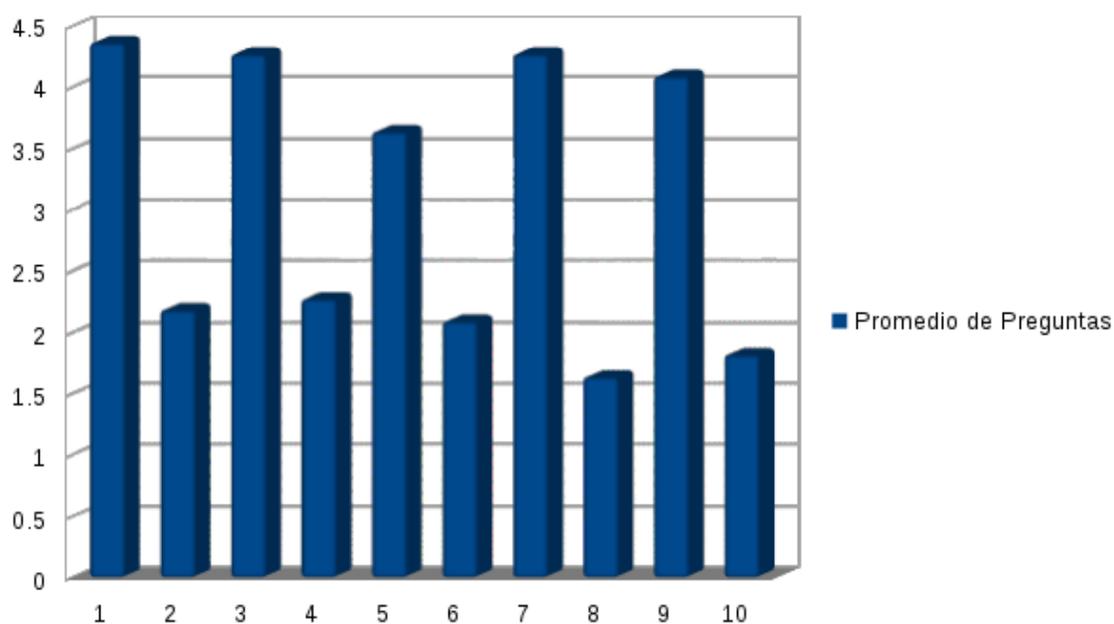


Figura 26. Gráfica de promedio de resultados por pregunta de la escala SUS del sistema MITS.

En la figura 26 se muestran los resultados promedio para cada uno de los reactivos en la escala SUS, las preguntas impares representan características positivas de la usabilidad del sistema, mientras que las preguntas pares se asocian a características de usabilidad negativas. De lo anterior resulta deseable obtener puntuaciones cercanas al 5

en los reactivos impares y cercana al uno en los pares. El reactivo positivo con mayor calificación promedio fue 1 promediando una calificación de 4.36 el cual corresponde a la pregunta “creo que me gustaría usar este sistema frecuentemente”. El reactivo negativo con la mayor puntuación promedio es el reactivo 4 correspondiente a la pregunta “creo que necesitaría soporte técnico para utilizar este sistema”. De lo anterior podemos concluir que el sistema presenta un alto grado de involucramiento por parte del usuario al fomentar el uso frecuente del sistema, sin embargo, es necesario en trabajar con la seguridad del usuario al utilizar el sistema de manera que no necesiten de soporte técnico para interactuar con MITS.

Los resultados muestran una capacidad de respuesta a las preguntas del usuario con un porcentaje elevado considerando que la interacción se dejó libre para los usuarios. Los errores de clasificación que tuvo el sistema en los casos en los que las preguntas del usuario si pertenecían a su dominio de conocimiento obedecen a variantes no consideradas en la forma de expresar las intenciones por parte del usuario y se pueden corregir fácilmente enriqueciendo el léxico y las expresiones de las intenciones del agente conversacional.

Otro punto interesante es el hecho de observar que tipo de interacción esperan los usuarios de un ITS para asistir al profesor en el uso de las herramientas de MOODLE, reflejado en aquellas preguntas que no se consideraron en el dominio del conocimiento inicial del sistema.

Estos temas se pueden incorporar fácilmente mediante la captura y retroalimentación de dicha información en la base de conocimiento del sistema.

Al finalizar la experimentación se realizó una entrevista con los usuarios quienes destacaron como factores positivos la facilidad de uso del sistema, la capacidad de reconocer variantes de lenguaje y la ventaja que ofrece al estar incrustado directamente en el entorno MOODLE. Las áreas de mejora que expresaron giraron en torno a hacer más llamativa la interfaz del usuario por medio de la inclusión de avatares, un ícono que representa a un usuario o algo que lo describa o referencia, el hecho de incorporar contenidos más complejos que los ofrecidos por las diferentes ayudas del sistema MOODLE. De igual manera los usuarios sugirieron dotar al sistema de la capacidad de revisar el contenido de los cursos para informar al profesor sobre el mismo. Aspectos como que tareas ya se entregaron, cuales hay pendientes y en general consultas acerca de sus cursos en la plataforma.

## **Capítulo 5. Conclusiones y trabajo futuro.**

Este capítulo tiene como propósito presentar el análisis del cumplimiento de los objetivos de la tesis, las conclusiones que se obtuvieron del trabajo con el sistema MITS así como las recomendaciones y líneas de trabajo futuro.

### **5.1 Análisis del cumplimiento de los objetivos.**

En el presente trabajo se describieron los elementos para la creación de un sistema de tutoría inteligente incorporado a un entorno LMS capaz de asistir a los profesores. Así mismo se realizó un caso de uso implementando el sistema MITS utilizando la arquitectura propuesta y evaluando la aceptación que tendría un tutor dedicado a ayudar a los profesores acerca del uso de las herramientas de MOODLE. Los objetivos analizados por esta tesis son los siguientes:

1. Analizar sistemas de gestión del aprendizaje disponibles y seleccionar uno que ofrezca herramientas robustas para la implementación de un curso en línea y permita la integración y desarrollo de módulos externos de manera que sea posible el desarrollo de un tutor inteligente integrado al mismo.

El primer objetivo consistió en el análisis de los sistemas de gestión del aprendizaje disponibles para la experimentación. En la sección 2.3 se realiza un análisis de las características teóricas de los mismos y se revisan los LMS disponibles actualmente. Se decidió utilizar el sistema MOODLE en su versión

3.1.4, debido a su gran difusión, el hecho de ser un sistema de código abierto, así como a la facilidad de integración con herramientas y plugins externos.

2. Determinar las mejores formas de asistir al profesor mediante el uso de un tutor inteligente, para desarrollar su conocimiento de las herramientas disponibles para la impartición de un curso en el sistema de gestión del aprendizaje elegido.

El segundo objetivo implicó la revisión de las técnicas empleadas por los ITS para asistir a los usuarios, lo cual se realiza en la sección 2.2. En esta sección se verificaron tanto las técnicas tradicionales como variedades de las mismas y nuevos enfoques de tutoría inteligente. Dado la madurez del usuario final (profesores universitarios) y las características variables de un tutor inteligente multi propósito, se eligió implementar un tutor inteligente conversacional e híbrido como se describe en las secciones 3.3 y 3.4. El agente propuesto empleó técnicas de tutoría de secuencia de currículo y tutoría basada en restricciones para asistir a los profesores en el uso de las herramientas de MOODLE.

3. Desarrollar un tutor inteligente integrado al sistema de gestión del aprendizaje seleccionado que implemente las formas adecuadas de asistir a los profesores en el uso de las herramientas dentro de un curso.

Para cumplir con el tercer objetivo se desarrolló el tutor inteligente MITS presentado en la sección 3.2 del presente documento. MITS es un tutor inteligente integrado al entorno MOODLE, el cual utilizó los principios de arquitectura

generales delineados a lo largo del documento implementando una función de similitud entre los usuarios y perfiles delineados a priori por el usuario.

El ITS se implementó por medio de un chatbot que utiliza una técnica de reconocimiento de patrones implementada por medio de un algoritmo de ventana deslizante con la distancia Levenshtein normalizada (Yujian y Bo, 2007) para encontrar las intenciones y entidades mencionadas por el usuario. Así mismo empleó un módulo de manejo de memoria capaz de recordar el estado de la conversación y seguir una secuencia lógica sobre los temas de los cuales estuvo hablando con el usuario.

4. Realizar los experimentos necesarios para verificar la efectividad de la propuesta, basados en cantidades suficientes de datos obtenidos a partir del uso de la propuesta por parte de profesores reales.

En la sección 4 se describen los experimentos realizados para verificar la usabilidad del chatbot, así como las impresiones generales de los usuarios al interactuar con la aplicación.

## **5.2 Conclusiones.**

Se obtuvieron resultados positivos a partir de la experimentación en los diferentes aspectos relacionados al presente trabajo de investigación.

El módulo para la obtención del perfil del usuario utilizando un vector de características generado a partir del uso de las herramientas de MOODLE por parte del

profesor se implementó comparando las medidas de similitud de coseno suave y la diferencia promedio normalizada. Ambas estrategias tuvieron un desempeño de clasificación de alrededor del 90% comparado con un humano experto, siendo ligeramente mejor en la clasificación la similitud de coseno suave. Lo anterior verifica que la estrategia utilizada es viable para la clasificación de usuarios de acuerdo a perfiles previamente proporcionados al sistema.

Cabe mencionar que la experimentación del módulo de clasificación de perfil se llevó a cabo de manera automatizada, recibiendo el sistema los perfiles a evaluar como parte de la configuración, por lo que un siguiente paso sería incorporar los resultados obtenidos por el proceso de minería de datos como reglas del sistema como se propone en la metodología y se ilustra en la figura 7.

El módulo de agente conversacional se implementó utilizando coincidencia de patrones mediante una medida de similitud de cadena basada en la distancia de Levenshtein normalizada (Yujian y Bo, 2007). Se obtuvo un 82.14% de eficiencia en la clasificación de intenciones y entidades cuando las preguntas realizadas por el usuario se circunscriben a la base de conocimientos del sistema. El margen de error se ocasionó por expresiones y ejemplos de intenciones y entidades que no habían sido cargados en el sistema al momento de la prueba, sin embargo pueden ser añadidos fácilmente a la base de conocimientos del sistema, lo cual presenta una poderosa ventaja en cuanto a la facilidad de extender el dominio de conocimiento en comparación a otros agentes conversacionales de reconocimiento de patrones desarrollados con AIML, puesto que no

requiere de un humano experto en el manejo de scripting para añadir conocimiento al chatbot.

La estrategia de búsqueda de patrones sintácticos implementada resultó con un excelente desempeño y en general la implementación del agente conversacional resultó exitosa y fácilmente extendible.

El módulo de tutoría híbrido se probó desde la perspectiva de los usuarios, quienes solicitaron la extensión de los contenidos del sistema, así como la posibilidad de conectarse a bases de información para mejorar sus técnicas de inferencia, sin embargo destacaron la facilidad de uso del sistema y consideraron que puede resultar en una herramienta útil tanto para la formación de profesores como para la interacción con sus cursos y tareas dentro de la plataforma.

El sistema obtuvo un desempeño aceptable en la escala SUS, indicando que el sistema puede ser adoptado fácilmente por una gran cantidad de usuarios.

Por último, la propuesta resultó interesante para los usuarios finales, lo cual indica que se tiene un gran margen de investigación y mejora tanto en los ITS conversacionales, como la integración entre ITS y LMS.

### **5.3 Recomendaciones y trabajo futuro**

El trabajo de investigación propone un marco metodológico y arquitectónico para el desarrollo de Sistemas de Tutoría Inteligente integrados a entornos LMS. En particular se implementó un Sistema de Tutoría Inteligente Conversacional para asistir a los profesores en el uso de las herramientas MOODLE el cual fue evaluado por profesores de

la UADY, por lo cual se pudo tener contacto directo con los potenciales usuarios del sistema. De las experiencias obtenidas con el presente trabajo de investigación se extraen las siguientes recomendaciones:

- Probar diferentes técnicas de tutoría. El marco arquitectónico desarrollado permite la implementación de módulos de manera independiente por lo que un estudio comparativo con diferentes técnicas de tutoría podría arrojar información interesante acerca de las mejores maneras de asistir a los profesores mediante ITS.
- Adaptar el chatbot a diferentes dominios de conocimiento. La experimentación en estas circunstancias permitirá generalizar los resultados obtenidos en la presente investigación y verificar si la estrategia conversacional implementada puede ser extendida a diferentes materias.
- Verificar la personalización del aprendizaje con datos reales de los profesores. La presente investigación realizó la validación de la clasificación de los perfiles de usuarios utilizando información generada de manera automatizada, sin embargo, la interacción con usuarios reales pudiera ayudar a corroborar que los resultados de clasificación obtenidos mediante las pruebas automáticas.
- Automatizar el proceso de recolección de perfiles iniciales mediante técnicas de agrupamiento de aprendizaje automático. Actualmente los perfiles de los usuarios del sistema se alimentaron de manera manual lo cual dificulta el proceso de adaptar los perfiles a la evolución del uso de la plataforma por

parte de los profesores. La implementación de estas técnicas no se han integrado aun en el software MITS, lo cual queda como un área de desarrollo a futuro.

- Retroalimentación automática de las expresiones de intenciones y entidades del sistema. El sistema podría mejorar su desempeño significativamente utilizando un proceso de retroalimentación que permita incluir expresiones no cargadas en la base de conocimiento utilizando las interacciones de los usuarios con el agente conversacional.
- Experimentar con diferentes medidas de similitud. La medida de similitud empleada en el sistema es sencilla y directa sin embargo, hay una gran cantidad de formas de calcular el perfil del usuario. Verificar varias técnicas en diferentes entornos resultaría útil para conocer en qué condiciones resulta mejor utilizar una técnica de clasificación sobre otra.
- Enriquecer los contenidos del sistema. Como parte de la investigación se verificó la viabilidad de elaborar un sistema de tutoría inteligente conversacional, por lo que los contenidos se limitaron a la ayuda que MOODLE ofrece respecto a sus herramientas, sin embargo, existe una gran cantidad de información que se podría emplear para asistir a los profesores de una manera más efectiva.
- Desarrollar estrategias para realizar la generación de la base de conocimientos del sistema de manera automatizada. Automatizar la extracción del conocimiento del sistema podría ahorrar significativamente tiempos de

implementación y facilitar el proceso de adquisición de conocimiento del sistema.

- Desarrollar módulos de conexión para consulta de APIs diversas. Lo anterior podría ampliar significativamente las capacidades del sistema para realizar diferentes procesos, además de proveer información mejorando la manera de asistir al usuario e integrando varias tareas desde una sola interfaz fácil de utilizar para los profesores.

#### **5.4 Divulgación de resultados.**

El trabajo de investigación propone un marco metodológico y arquitectónico para el desarrollo de Sistemas de Tutoría Inteligente integrados a entornos LMS. En particular este trabajo de investigación propone un marco metodológico y arquitectónico para el desarrollo de Sistemas de Tutoría y la implementación del sistema MITS para asistir a los profesores en el uso de las herramientas de MOODLE. Con el fin de presentar resultados preliminares, la propuesta y metodología de trabajo se divulgaron los resultados en:

Campos, M., Menéndez, V.H., Zapata, A. (2017). MITS: Sistema de Tutoría Inteligente Conversacional para Asistir a los Profesores en el Uso de las Herramientas de MOODLE. En Prieto, M., Pech, S., Zapata, A. Tecnología y Aprendizaje- Avances en el mundo académico hispano, pp. 522-529. ISBN: 978-84-697-2772-0.



### Lista de referencias

Agrawal, R., Imieliński, T., Swami, A. (1993). Mining association rules between sets of items in large databases. Proceedings of the 1993 ACM SIGMOD international conference on Management of data - SIGMOD '93 (Vol. 22, pp. 207–216). New York, New York, USA: ACM Press. Doi:10.1145/170035.170072

Aimeur, E., Frasson, C. y Dufort, H. (2000). Cooperative learning strategies for intelligent tutoring systems. *Applied Artificial Intelligence*, 14(5), 465-489.

Alavi, M., Leidne, D. (2001). Review: Knowledge Management and Knowledge Management Systems: Conceptual Foundations and Research Issues. *Management Information Systems Research Center, University of Minnesota*, 25(1), 107–136.

Alvi, A. y Ali M. (2010). Revival of Tutor Model: A Domain Independent Intelligent Tutoring System. *BVICAM's International Journal of Information Technology*. 2(1), 50-52.

Ahmad, J., Gea, M., Paderewski, P., y Gutierrez, F. (2005). A comparison and evaluation Of Open source learning management system. *IADIS International Conference- Applied Computing 2005*. Recuperado el 24 de agosto del 2015 de: <https://MOODLE.org/pluginfile.php/554/.../Jamil1.pdf>

Apereo Foundation. (2016). Sakai. Recuperado de: <https://sakaiproject.org/>

Bangor, A., Miller, J., Kortum, P., “Determining What Individual SUS Scores Mean: Adding an Adjective Rating Scale” *Journal of Usability Studies*, vol. 4, issue 3, pp. 114-123, Mayo 2009.

Bates A. (2000). Distance education in dual mode higher education institutions: Challenges and changes. Retrieved December 27, 2005 from <http://bates.cstudies.ubc.ca/papers/challenizesandchanges.html>.

Beal, C. R., Beck, J., y Woolf, B. (1998). Impact of intelligent computer instruction on girls' math self concept and beliefs in the value of math. Paper presented at the annual meeting of the American Educational Research Association.

Berdun, L., Amandi, A., y Campo, M. (2014). An intelligent tutor for teaching software design patterns. *Computer Applications In Engineering Education*, 22(4), 583-592.

Bergin, R. A. y Fors, U. G. H. (2003). Interactive Simulated Patient – An advanced Tool for Student-activated Learning in Medicine and Healthcare. *Computers & Education*, 40(4), 361-376. doi:10.1016/S0360-1315(02)00167-7.

Berz, M, Erdelyi B., y Hoefkens, J. (1999). Experiences with interactive remote graduate instruction in beam physics. *Journal of Interactive Learning Research*, 10(1), 49-58.

Blackboard (2016). Recuperado el 27 de abril de 2016 de: <http://www.blackboard.com/>

Biolchini, J., Mian, P.G., Natali, A.C., Travassos, G.H. (2005). Systematic Review in Software Engineering. Technical Report ES 679/05.

Bonnet, A. (1985), *Artificial Intelligence; Promise and Performance*. London: Prentice Hall.

Brogan, P. (1999). Using the Web for Interactive Teaching and Learning. Macromedia white paper, 1–38.

Brooke, J. (1996) “SUS - A quick and dirty usability scale”, Redhatch Consulting Ltd, United Kingdom

Bulut Özek, Müzeyyen, Zuhtu Hakan Akpolat, and Ahmet Orhan. 2013. "A web-based intelligent tutoring system for a basic control course." *Computer Applications In Engineering Education* 21, no. 3: 561-571. *Applied Science & Technology Source*, EBSCOhost (accessed April 19, 2016).

Camacho, P., Zapata, A. Menéndez, V.H., Canto, P.J. (2017). Análisis sobre el desempeño de los profesores en el Sistema de Gestión del Aprendizaje UADY Virtual. En Prieto, M., Pech, S., Zapata, A. *Tecnología y Aprendizaje- Avances en el mundo académico hispano*, pp. 368-375. ISBN: 978-84-697-2772-0.

Carbonero, E., Morato, J. “Folsonomía” 2009 [Online]. Disponible en: <http://glossarium.bitrum.unileon.es/Home/folksonomia-folksonomy> [Accedido 8 de Junio de 2017]

Casany, M. m., Alier, M. m., Mayol, E. m., Piguillem, J. j., Galanis, N. n., García-Peñalvo, F. f., y Conde, M. m. (2012). Moodbile: A Framework to Integrate m-Learning Applications with the LMS. *Journal Of Research & Practice In Information Technology*, 44(2), 129-149.

Curilem, S. G., Barbosa, A. R., y de Azevedo, F. M. (2007). Intelligent tutoring systems: Formalization as automata and interface design using neural networks. *Computers & Education*, 49(3), 545-561. doi:10.1016/j.compedu.2005.10.005

Dāboliņš, J. y Grundspenķis, J. (2013). The Role of Feedback in Intelligent Tutoring System. *Applied Computer Systems*, 14(1), 88-93.

De Praetere, T., Lobelle, C. (2010). The Dokeos e-Learning Project Management Guide (pp. 2007–2010).

Dempster, A.P, Laird, N.M, Rubin, D. (1977). Maximum Likelihood from Incomplete Data via the EM Algorithm. *Journal of the Royal Statistical Society. Series B (Methodological)*, 39(1), 1–38.

Dogan, B. y Camcuru, A.Y. (2010). Visual clustering of mulidimensional educational data from an intelligenet tutoring system. *Computer Applications in engineering Education*. 18(2), 375-382.

Dougiamas, M., Taylor, P.C. (2003). Moodle : Using Learning Communities to Create an Open Source Course Management System. *Proceedings of the World Conference on Educational Multimedia, Hypermedia and Telecommunications* (pp. 171–178.).

Downes, S. (2005). Feature: E-learning 2.0. *eLearn Magazine*, ACM Press, 10.

DreamBox (2016). DreamBox Learning. Recuperado el 02 de mayo 2016 de: <http://www.dreambox.com/>.

Duchastel, P., Imbeau J. (1988). Intelligent Computer Assisted Instruction (ICAI): Flexible Learning Through Better Student Computer Interaction. 12th International Conference on Improving University Teaching, July 1986, Heidelberg, Germany.

Edwards A.L. (1976). *Introduction to Linear Regression and Correlation*. (A series of books in psychology.) W. H. Freeman and Comp., San Francisco.

Eiben, A. E., Smith, J. (2003). What is an Evolutionary Algorithm? *Introduction to Evolutionary Computing* (pp. 15–35).

Eliot, C., y Woolf, B. (1994). Reasonong about the user within a simulation-based-real-time training system. In *Proceedings of the fourth international conference on user modeling*, 121-126.

Estrada, P., M., (2016). Definition of a data model that includes the instructional design to describe the elements involved in the design of virtual courses. *Sistemas & Telemática*, vol. 14, núm. 36, 2016, pp. 59-68

Fayyad, U., Piatetsky-shapiro, G., Smyth, P. (1996). From Data Mining to Knowledge Discovery in, 17(3), 37–54.

Ferreira-Satler, M., Romero, F., Menendez-Dominguez, V. y Zapata, A. (2012). Fuzzy ontologies-based user profiles applied to enhance e-learning activities. *Soft Computing* 16 (7), 1129-1141

Fidelis. (2016). Fidelis Inc. Recuperado el 16 de mayo 2016 de: <http://www.fideliseducation.com/>

Fisher, D.H. (1987). Knowledge Acquisition Via Incremental Conceptual Clustering. *Machine learning*, 2(1980), 139–172.

Fishtree. (2016). Fishtree Inc. Recuperado el 27 de abril de 2016 de: <https://www.fishtree.com/>

FMAT-UADY. (2015). Facultad de Matemáticas UADY. Recuperado 1 de enero de 2015, a partir de <http://www.matematicas.uady.mx>

Francesa, A., Espinoza, J. y Chacon, M. (2012). Hacia una Herramienta para el Diseño Instruccional en Educación Superior. Conferencia Ibérica de Sistemas y Tecnologías de la Información 2012. Madrid.

García, E., Romero, C., Ventura, S., De Castro, C. (2007). An architecture for making recommendations to courseware authors using association rule mining and collaborative filtering, (Dm).

García-Peñalvo, F. J. y Seoane-Pardo, A. M. (2015). Una revisión actualizada del concepto de e-learning. Décimo Aniversario. *Education on Knowledge Society (EKS)*, 16(1), 119-144. <http://dx.doi.org/10.14201/eks2015161119144>

Graesser, A. C., Wiemer-Hastings, K., Wiemer-Hastings, P., Kreuz, R., y TRG. (1999). AutoTutor: A simulation of a human tutor. *Journal of Cognitive Systems Research*, 1, 35-51, doi:10.1016/S1389-0417(99)00005-4

Graesser, A., Chipman, P., Haynes, B. y Olney, A. (2005). Autotutor: an intelligent tutoring system with mixed-initiative dialogue, *IEEE Transactions on Education* 48 (2005) 612–618.

Graesser, A. C. (2016). Conversations with AutoTutor Help Students Learn. *International Journal of Artificial Intelligence in Education*, vol. 26, no. 1, pp. 124–132, 2016.

Gordillo, M., E., y Chávez Blando M. (2007). Diseño Instruccional, elemento clave en el desarrollo de cursos para Ambientes Virtuales de Aprendizaje. México: Dirección de Tecnología Educativa, IPN.

Growth Engineering. (2016). Recuperado el 27 de abril de 2016 de: <http://www.growthengineering.co.uk/>

Haake, M. y Gulz A. (2009). A look at the roles of look & roles in embodied pedagogical agents-a user preference perspective. *International Journal of Artificial Atitifivial Intelligence in Education*, vol. 19, pp. 39–71, 2009.

Hafdi, M. y Bensebaa, T. (2013). Design and evaluation of a intelligent tutoring system by expert system. *Intelligent Decision Technologies*, 7(4) , (pp 253-264).

Hernández, J., Ramírez, M.J., Ferri, C. (2004). *Introducción a la Minería de Datos*. Pearson Prentice Hall.

Hosmer, D.W., Stanley, L. (2000). *Applied Logistic Regression*. (John Wiley & sons, Ed.).

Hyacinth, S. N. (1990). Intelligent Tutoring Systems: an overview. *Artificial Intelligence Review*, 4, 251-277.

Itmazi, J. (2005). *Sistemas flexibles de gestión del E-Learning para soportar el aprendizaje en las universidades tradicionales y abiertas*.

Kapp, K.M. (2003). *Five Technological Considerations When Choosing an E-Learning Solution*. eLearn Magazine, ACM Press, 7.

Kearsley, G. (1987). *Artificial intelligence and instruction applications and methods* (p. 158). Reading, MA: Addison-Wesley Publishing.

Keles, A., y Keles, A. (2011). Intelligent Tutoring Systems: Best Practices. In S. Stankov, V. Glavinic, & M. Rosic (Eds.) *Intelligent Tutoring Systems in E-Learning Environments: Design, Implementation and Evaluation* (pp. 1-26). Hershey, PA: Information Science Reference. doi:10.4018/978-1-61692-008-1.ch001

Keles, A., Ocak, R., y Gülcü, A. (2009). ZOSMAT: Web-based intelligent tutoring system for teaching-learning process. *Expert Systems with Applications*, 36, 1229-1239. doi:10.1016/j.eswa.2007.11.064,

Kerka. S. (1996). *Distance Learning the internet and theWorld wide web*. ERIC Digtst. Columbus. OH: ERIC Clearinshouse on Adult Career and Vocational Education. (ERIC Identifier. ED395214).

Kerly, A., Ellis R. y Bull, S. (2008). CALMsystem: A Conversational Agent for Learner Modelling. *Knowledge-Based Systems*, vol. 21, no. 3, pp. 238–246, 2008.

- Khan, B. (1997). Web-based Instruction (WBI): What is it and Why is it? In Khan, B. (ed), Web-based Instruction, Educational Technology Publications.
- Kitchenham, B., Charters, S. (2007). Guidelines for performing Systematic Literature Reviews in Software Engineering, Version 2.3. Keele University and Durham University, EBSE Technical Report, EBSE-2007-01: UK.
- Kohonen, T. (1988). Self-organized formation of topologically correct feature maps. In E. Anderson, J.A. Rosenfeld (Ed.), *Neurocomputing: foundations of research* (pp. 509–521).
- Kozierkiewicz, A. H. y Nguyen, N. T. (2011). A method for learning scenario determination and modification in intelligent tutoring systems. *International Journal of Applied Mathematics & Computer Science*. 21(1), 69-82.
- Künzel, J. y Hämmer, V. (2003). Simulation in University Education: The Artificial Agent PSI as Teaching Tool. *Simulation – Special Issue on Modeling and Simulation in Teaching and Training*, 82, 11. 761-767.
- Lajoie, S. F., y Lesgold, A. (1989). Apprenticeship training in the workplace: Computer-coached practice environment as a new form of apprenticeship. *Machine-Mediated Learning*, 3, 7-28.
- Latham, A., Crockett, K., McLean D. y Edmonds, B. (2012). A conversational intelligent tutoring system to automatically predict learning styles. *Computers and Education*, vol. 59, no. 1, pp. 95–109, 2012. [Online]. Available: <https://www.scopus.com/inward/record.uri?eid=2s2.084863108208&doi=10.1016>
- Latham, A., Crockett, K., McLean D. y Edmonds, B. (2012). A conversational intelligent tutoring system to automatically predict learning styles. *Computers and Education*, vol. 59, no. 1, pp. 95–109, 2012. [Online]. Available: <https://www.scopus.com/inward/record.uri?eid=2s2.084863108208&doi=10.1016>
- Latham, A., Crockett, K., McLean D. (2014). An Adaptation Algorithm for an Intelligent Natural Language Tutoring System. *Computers & Education*, vol. 71, pp. 97–110, 2014. [Online]. Available: <http://dx.doi.org/10.1016/j.compedu.2013.09.014>
- Lavandelis, E. y Bicans, J. (2011). Multi-Agent and Service Oriented Architectures for Intelligent Tutoring Systems. *Scientific Journal of Riga Technical University*. 47 (pp 11-20).
- Lease, A. J., y Brown, T. A. (2009). DISTANCE LEARNING PAST, PRESENT AND FUTURE. *International Journal Of Instructional Media*, 36(4), 415-426.

Lee, C. y Sapiyan M. B. (2006). Bayesian and Fuzzy Logic Student Model in the C++ STL Intelligent Tutoring System. *International MultiConference of Engineer & Computer Scientist* 2006. 98-103.

Lerís-López, D., Veá-Muniesa, F. y Velamazán-Gimeno, A. (2015). Adaptive learning in Moodle: three practical cases. *Education on Knowledge Society (EKS)*, 16(4), 138-157.

López, J., M., Romero, E., y Roperó, E. (2010). Utilización de Moodle para el desarrollo y evaluación de competencias en los Alumnos Formación Universitaria, vol. 3, núm. 3, 2010. Centro de Información Tecnológica. La Serena, Chile

Lynch, T.D., Lynch, C.E. (2002). Web-Based Education. *The Innovation Journal: The Public Sector Innovation Journal*, 8(4), 1–28.

Macqueen, J. (1967). Some Methods for Classification and Analysis of Multivariate Observations. *Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability*, 1, 281–297.

Martens, A. y Harrer, A. (2011). Intelligent Tutoring System Architecture Rebuilt: A Pattern Approach. In S. Stankov, V. Glavinic, & M. Rosic (Eds.) *Intelligent Tutoring Systems in E-Learning Environments: Design, Implementation and Evaluation* (pp. 1-26). Hershey, PA: Information Science Reference. doi:10.4018/978-1-61692-008-1.ch001

Martens, A. y Himmelspach, J. (2005). Combining Intelligent Tutoring and Simulation Systems. In *Proceedings of the International Conference on Simulation in Human Computer Interfaces, SIMCHI, part of the Western Multi Conference WMC '05*. New Orleans, LA (pp 65-70).

McIssac, M. y Gunawardena, C. (1996). Distance education. In D. H. Jonassen (Ed.). *Handbook of research for educational communications and technology*. New York: Simon and Shuster.

Mikic Fonte, F., Burguillo, J. y Nistal, M., (2009). TQ-Bot: An AIML-based tutor and evaluator bot, *Journal of Universal Computer Science* 15 1486–1495.

Mikic Fonte, F., Burguillo, J. y Nistal, M. (2012). An intelligent tutoring module controlled by BDI agents for an e-learning platform. *Expert Systems with Applications*, vol. 39, no. 8, pp. 7546–7554, 2012. [Online]. Available: <https://www.scopus.com/inward/record.uri?eid=2s2.84857655173&doi=10.1016%2Fj.eswa.2012.01.161&partnerID=40&md5=b22cbbf1d53d5ec0ddf238f4900baf31>

Moisa, V. (2013). Adaptive Learning Management System. *Journal of Mobile , Embedded & Distributed Systems*. 5(2), 70-77.

Moodie, P. P., y Kunz, P. (2003). Recipe for an Intelligent Learning Management System (ILMS). In *11th International Conference on Artificial Intelligence in Education*. Conference held at Sydney, Australia.

National Center for Education Statistics (2002). Learning resources and technology: Educational technology. Retrieved March 2, 2006 from <http://nces.ed.gov/pubs2003/digest02/ch7.asp>.

O’Shea, J., Bandar Z. y Crockett, K. (2011). Systems engineering and conversational agents,” in *Intelligence-based systems engineering*, . L. C. J. E. A. Tolk, Ed. Berlin Hedelberg: Intelligent Systems Reference Library 10: Springer-Verlag, 2011.

Pearson education Inc. (2016). eCollege. Recuperado el 11 de mayo de 2016 de: <http://www.ecollege.com/index.php>

Pek, P. K y Poh, K. L. (2005). Making Decisions in an Intelligent Tutoring System. *International Journal of Information Technology & Decision Making*. 4(2), (pp 207-233).

Pilato, G. , Pirrone, R. y Rizzo, R. (2008). A KST-based system for student tutoring. *Applied Artificial Intelligence*, vol. 22, no. 4, pp. 283–308, 2008.

Polo, M. (2001). El diseño instruccional y las tecnologías de la información y la comunicación. Extraído el 31 de Enero, 2006 del sitio web Universidad Central de Venezuela: <http://www.revele.com.ve/programas/indice/ria.php?id=12093&rev=docencia>

Quinlan, J R. (2007). Induction of Decision Trees. *Machine learning*, 1(1), 81–106.

Quinlan, J.R. (1993). C4.5 : Programs for Machine Learning (Vol. 240, pp. 235–240).

Rištofič, A.K. (2005). Recommender System for Adaptive Hypermedia Applications. *Proc. Stud. Res. Conf. Informat. Inf. Technol.* (pp. 229–234).

Romero, C., Ventura, S. (2007). Educational data mining: A survey from 1995 to 2005. *Expert Systems with Applications*, 33(1), 135–146. doi:10.1016/j.eswa.2006.04.005

Romero, C., Ventura, S. y Zafra, A., De Bra, P. (2009). Applying Web usage mining for personalizing hyperlinks in Web-based adaptive educational systems. *Computers & Education*, 53(3), 828–840. doi:10.1016/j.compedu.2009.05.003

Rosenberg, M. J. (2001). *E-learning: Strategies for delivering knowledge in the digital age*. Columbus, OH: McGraw-Hill.

Rossett, A., y Sheldon, K. (2001). *Beyond the podium: Delivering training and performance to a digital world*. San Francisco, CA: Jossey-Bass/Pfeiffer.

Ruiz, F., Polo, M. y Piattini, M. (2002). Utilización de Investigación-Acción en la Definición de un Entorno para la Gestión del Proceso de Mantenimiento del Software. 1er. Workshop en: Métodos de Investigación y Fundamentos Filosóficos en Ingeniería del Software y Sistemas de Información.

Rumelhart, D.E., Hinton, G.E., Williams, R.J. (1986). Learning internal representations by error propagation. In J. L. Rumelhart, David E. and McClelland (Ed.), *Parallel distributed processing: explorations in the microstructure of cognition* (pp. 318–362).

Sánchez, J.G., Rosas, A.C., Ricardo, J., Terrats, A. (2006). en México: un primer acercamiento ( 1 ). *Virtual educa* (pp. 1–17).

Sánchez, J. (2009). Plataformas de Enseñanza Virtual para Entornos Educativos. *Pixel-Bit: Revista de Medios y Educación*, 217-233.

Scheffer, T. (2004). Finding Association Rules that Trade Support Optimally Against Confidence. *Intell. Data Anal.*, 9(4), 381–395.

Serce, F. C., Alpaslan, F. N, y Jain, L. C. (2008). Intelligent learning system for online learning. *International Journal of Hybrid Intelligent Systems*. 5(3) 129-141.

Sidorov, G., Gelbukh, A., Gómez-Adorno, H., Pinto, D. (2014). Soft Similarity and Soft Cosine Measure: Similarity of Features in Vector Space Model. *Computación y Sistemas* 18 (3): 491-504. doi:10.13053/CyS-18-3-2043.

Skillsoft Company (2016). Recuperado el 27 de abril de 2016 de: <http://www.sumtotalsystems.com/>

Smart Parrow. (2016). Smart Parrow PTY LTD. Recuperado el 02 de mayo 2016 de: <https://www.smartparrow.com/>

Son, L. (2016). Dealing with the new user cold-start problem in recommender systems: A comparative review. *Information Systems Vol 58*. pp(87-104).

Stankov, S., Glavinic, V., y Rosic, M. (2011). *Intelligent Tutoring Systems in E-Learning Environments: Design, Implementation and Evaluation* (pp. 1-446). Hershey, PA: IGI Global. Doi:10.4018/978-1-61692-008-1

Stankov, S., Rosic, M., Zitko, B y Grubisic, A. (2008). Tex-Sys model for building intelligent tutoring systems. *Computers & Education*. 51(3), 1017-1036.

Tegos, S., Demetriadis, S. y Karakostas, A. (2015). Promoting academically productive talk with conversational agent interventions in collaborative learning settings. *Computers & Education*, vol. 87, pp. 309–325, sep 2015. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/S0360131515300191>

Tobón, M. (2007). *Diseño Instruccional en un Entorno de Aprendizaje Abierto*. Universidad Tecnológica de Pereira.

Toronto University. (2002). ATutor: Learning Management System. Retrieved September 25, 2012, from <http://atutor.ca/atutor/>

Tullis, T., Stetson, J. (2006). “A Comparison of Questionnaires for Assessing Website Usability” Human Interface Design Department, Fidelity Center for Applied Technology, Boston, MA, 2006

Traxler, J. (2005). Defining mobile learning. *IADIS International Conference Mobile Learning 2005* (pp. 261–266).

Tsai, S., Machado, P. (2002). *E-Learning Basics: Essay: E-learning, online learning, web-based learning, or distance learning: unveiling the ambiguity in current terminology*. eLearn Magazine, ACM Press, 3.

UADY (2010). *Plan de Desarrollo Institucional 2010-2020*. Obtenido de Universidad Autónoma de Yucatán: <http://www.pdi.uady.mx/docs/pdi.pdf>

UADY (2012). *Modelo Educativo para la Formación Integral*. Obtenido de Universidad Autónoma de Yucatán: [http://www.dgda.uady.mx/media/docs/mefi\\_dgda.pdf](http://www.dgda.uady.mx/media/docs/mefi_dgda.pdf)

UADY (2013). *Plan de Desarrollo UADY Virtual 2013-2020*. Obtenido de: [http://uadyvirtualcloud.uady.mx/pluginfile.php/142/mod\\_resource/content/2/Plan%20Desarrollo%20UADY%20Virtual%202013-2020.pdf](http://uadyvirtualcloud.uady.mx/pluginfile.php/142/mod_resource/content/2/Plan%20Desarrollo%20UADY%20Virtual%202013-2020.pdf)

UADY. (2014). *Plan de Desarrollo Institucional, 2014-2022*. Recuperado de: <http://www.pdi.uady.mx/docs/pdi.pdf>

University of cologne. (2004). Didactical concept of ILIAS. Retrieved November 2, 2012, from [http://www.ilias.de/docu/goto\\_docu\\_lm\\_392.html](http://www.ilias.de/docu/goto_docu_lm_392.html)

Urban-Lurain, M. (2003). An historic review in the Context of the Development of Artificial Intelligence and Educational Psychology. Retrieved October 24, 2009, from <http://www.cse.msu.edu/rgroups/cse101/ITS/its.htm>

Urquijo, C., y Tapia, H. (2007). El Diseño Instruccional como Base para la Construcción de Entornos de Aprendizaje. In J. Figueroa, E. Vargas, & N. Cruz (Eds.) Metodología para la Educación a Distancia (pp. 85-94).

usability.gov, (2017). “System Usability Scale (SUS)”, [Online] Disponible en: <https://www.usability.gov/how-to-and-tools/methods/system-usability-scale.html> [Accedido 2 de Junio de 2017]

Valentine D. (2002) Distance learning: Promises problems and possibilities. Online Journal of Distance Learning Administration 5 (3). Retrieved December 29 32005 from <http://www.westga.edu/-distance/ojdl/fall53/valentine53.html>.

Wegner, E. (1987). Artificial intelligence and tutoring systems, Los Altos, CA: Meagan Kaufmann

Weerasinghe, A. ; Mitrovic, A.; Martin, B.: Towards individualized dialogue support for ill-defined domains, *International Journal of Artificial Intelligence in Education*. Vol. 19, No. 4, pp. 357-379 (2009).

Wu, C. (1999). Teaching Rankine cycle by using an intelligent computer-aided instruction software. *International Journal Of Mechanical Engineering Education*, 27(4), 337.

Yazdani, M. (1987). Intelligent tutoring systems survey. *Artificial Intelligence Review*, 1, 43-52. doi:10.1007/BF01988527.

Ye, P., Baldwin, T. (2006). Semantic Role Labelling of Prepositional Phrases, 5(3), 228–244.

Yujian L., Bo L. (2007). A Normalized Levenshtein Distance Metric. *IEEE Transactions On Pattern Analysis And Machine Intelligence*. Vol 29. No. 6, 1091 – 1095-

Zapata, A., Menéndez, V., Prieto, M. y Romero C. (2013). A framework for recommendation in learning object repositories: An example of application in civil engineering. *Advances in Engineering Software* Vol 56, pp. 1-14

Zhang, H. (2004). The Optimality of Naive Bayes. In Z. Barr, V., Markov (Ed.), Proceedings of the Seventeenth International Florida Artificial Intelligence Research Society Conference.