

UNIVERSIDAD AUTÓNOMA DE BAJA CALIFORNIA
FACULTAD DE CIENCIAS
DOCTORADO EN CIENCIAS



ARQUITECTURA NÚCLEO PARA LA COOPERACIÓN ENTRE
TUTORES INTELIGENTES
TESIS

QUE PRESENTA PARA OBTENER EL GRADO DE DOCTOR EN
CIENCIAS

PRESENTA:
RODOLFO ALAN MARTÍNEZ RODRÍGUEZ

DIRECTORES DE TESIS:
DR. OMAR ÁLVAREZ XOCHIHUA
DR. JOSÉ ÁNGEL GONZÁLEZ FRAGA

ENSENADA, BAJA CALIFORNIA, MÉXICO, ENERO DEL 2020



UNIVERSIDAD AUTÓNOMA DE BAJA CALIFORNIA

FACULTAD DE CIENCIAS

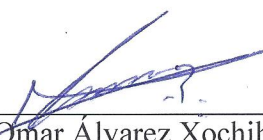
Arquitectura núcleo para la cooperación entre tutores

TESIS

Que para obtener el grado de Doctor en Ciencias presenta:

Rodolfo Alan Martínez Rodríguez

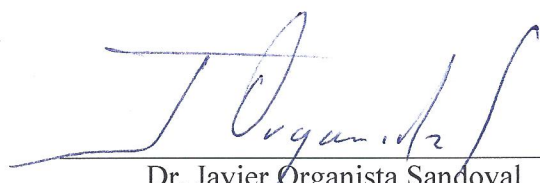
Aprobada por:



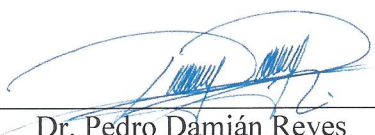
Dr. Omar Álvarez Xochihua
Director de Tesis



Dr. José Ángel González Fraga
Co-Director de Tesis



Dr. Javier Organista Sandoval
Miembro del Comité



Dr. Pedro Damián Reyes
Miembro del Comité



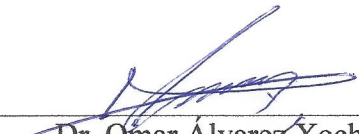
Dra. Olivia Denisse Mejía Victoria
Miembro del Comité

Ensenada Baja California, México, Enero del 2020

RESUMEN de la Tesis de **Rodolfo Alan Martínez Rodríguez**, presentada como requisito parcial para la obtención del grado de DOCTOR EN CIENCIAS del programa Maestría y Doctorado en Ciencias e Ingeniería (MYDCI), de la Universidad Autónoma de Baja California. Ensenada, Baja California, México. Enero del 2020.

Arquitectura núcleo para la cooperación entre tutores inteligentes

Resumen aprobado por:



Dr. Omar Álvarez Xochihua
Director de Tesis



Dr. José Ángel González Fraga
Co-Director de Tesis

La arquitectura de software propuesta, orientada a homogeneizar el modelo del estudiante, atiende una problemática con alcance internacional que enfrentan los desarrolladores de sistemas de tutoría inteligente. Este trabajo de tesis presenta el diseño, implementación y evaluación de una arquitectura de software que permite la comunicación entre tutores inteligentes, permitiendo homogeneizar el modelo del estudiante para ser compartido entre tutores inteligentes de diferentes dominios. Esta investigación pretende ampliar el desarrollo de arquitecturas de software actuales que se apoyan a través de plataformas digitales de aprendizaje para transmitir conocimiento y asesoría.

Para el modelado del conocimiento del usuario se utilizaron técnicas de modelado mediante semántica ontológica. El diseño de la arquitectura integral de tutores inteligentes se fundamentó en un metamodelo basado en la *Arquitectura Dirigida por Modelos*, separando los flujos funcionales (servicios) de las tecnologías que los implementan. Al

mismo tiempo, se sustenta una arquitectura de comunicación multiagentes deliberativa, mediante una relación positiva e interacción implícita. La característica deliberativa de la arquitectura propuesta se sustentó en el modelado de conocimiento basado en ontologías, y una máquina de inferencia basada en reglas. El protocolo de comunicación entre tutores implementado se materializó con la especificación xAPI y un repositorio LRS, debido a las bondades que estos ofrecen para que sistemas muy diferentes puedan comunicarse entre sí mientras interactúan con diferentes tecnologías y contenidos.

Para realizar la evaluación del desempeño de la arquitectura implementada se utilizó una metodología cuantitativa, considerando que este estudio se basó en aspectos observables y medibles en nuestro entorno real. La evaluación de la arquitectura propuesta se dividió en 2 etapas. Durante su desarrollo, en la primera etapa, la arquitectura fue evaluada por 10 grupos de estudiantes de la Facultad de Ciencias Administrativas y Sociales de la Universidad Autónoma de Baja California unidad Valle Dorado, que ingresaron en el periodo 2018-1 en el área económica administrativa. El estudio conformado por alrededor de 400 alumnos, consistiendo del 100% del universo de ingreso, permitió realizar las pruebas de calidad de funcionalidad de software consideradas en la investigación: operatividad, rendimiento, compatibilidad y escalabilidad. Durante la prueba de operatividad se obtuvo un acceso concurrente máximo de 106 peticiones simultáneas, donde el sistema soportó las peticiones sin interrumpir sus servicios y sin degradar su tiempo de respuesta. Se observó que el protocolo de comunicación utilizado, la especificación xAPI y el repositorio LRS, permiten que la interacción entre tutores y cada uno de sus componentes se lleve a cabo de manera correcta y transparente al usuario.

En la segunda etapa, con el objetivo de obtener información genérica para el modelo del estudiante, se contó con la participación de 141 alumnos para determinar el nivel de autoestima de acuerdo al instrumento inventario de autoestima de Coopersmith, bajo las categorías Familiar, Escolar y General. Con lo anterior se generó la terna FEG. Adicionalmente, 114 alumnos participaron para determinar el estilo de aprendizaje de acuerdo al instrumento de Bandler y Ginder, enfocado en el modelo Visual, Auditivo y Kinestésico Generando la terna VAK.

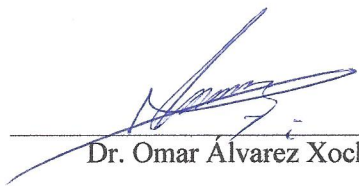
Para medir la aceptación y la eficiencia de la tutoría personalizada brindada por la arquitectura propuesta se realizó una encuesta adaptada del instrumento de usabilidad y experiencia de usuario UMUX-Lite, validada por un experto en psicología pedagógica. El resultado de este estudio mostró un alto nivel de aceptación en la asesoría personalizada ofrecida a estudiantes por la arquitectura de software propuesta y el ambiente integral. Con el diseño e implementación de la arquitectura propuesta, los resultados obtenidos, el análisis detallado de los resultados de las evaluaciones y una discusión de lo encontrado, se pudo determinar las conclusiones de las aportaciones del estudio, líneas de investigación factibles de explorar y trabajo futuro del grupo de investigadores participantes en el estudio.

Palabras clave: Arquitectura de software, Ontologías, xAPI, Sistemas de Tutoría Inteligente

RESUMEN de la Tesis de **Rodolfo Alan Martínez Rodríguez**, presentada como requisito parcial para la obtención del grado de DOCTOR EN CIENCIAS del programa Maestría y Doctorado en Ciencias e Ingeniería (MYDCI), de la Universidad Autónoma de Baja California. Ensenada, Baja California, México. Enero del 2020.

Core architecture for cooperation between intelligent tutor system

Resumen aprobado por:



Dr. Omar Álvarez Xochihua
Director de Tesis



Dr. José Ángel González Fraga
Co-Director de Tesis

The proposed software architecture, focused on homogenizing the student model, answers an international research problem, faced by developers of intelligent tutoring systems. This thesis work presents the design, implementation, and evaluation of a software architecture that allows communication between intelligent tutors, allowing the student's model to be homogenized to be shared between intelligent tutors from different domains. This research intends to expand the development of current software architectures that are supported through digital learning platforms to transmit knowledge and mentoring.

For the modeling of user knowledge, semantic web tools were used; The decision to use techniques of knowledge modeling through ontological semantics was chosen. The design of the integral architecture of intelligent tutors was based on a metamodel based on the Model-Driven Architecture, separating the functional flows (services) of the technologies that implement them. At the same time, it is sustained a deliberative multi-

communication architecture, through a positive relationship and implicit interaction. The deliberative feature of the proposed architecture was based on ontologies knowledge modeling and a rule-based inference engine. The communication protocol between tutors implemented was materialized with the xAPI specification and a LRS repository, due to the benefits they offer. So that, very different systems can communicate with each other while interacting with different technologies and contents.

To achieve the performance evaluation of the implemented architecture, a quantitative methodology was used, considering that this study was based on observable and measurable aspects in our real environment. The evaluation of the proposed architecture was divided into 2 stages. During its development, in the first stage the architecture was evaluated by 10 groups of students of the Facultad de Ciencias Administrativas y Sociales campus Valle Dorado (Spanish acronym for Faculty of Administrative and Social Sciences), who entered during 2018-1 period in the administrative-economic area. The survey made up of around 400 students, consisting of 100% of the total admission, allowed to perform quality tests of the software functionality considered in the research: interoperability, performance, compatibility, and scalability. During the operability test, maximum concurrent access of 106 simultaneous requests was obtained, where the system supported the requests without interrupting its services and without degrading its response time. It was observed that the communication protocol used, the xAPI specification and the LRS repository, allow the interaction between tutors and each of its components to be carried out correctly and transparently to the user.

In the second stage, in order to obtain generic information to build the student model, 141 students participated to determine the level of self-esteem according to the Coopersmith self-esteem inventory instrument, under the Family, School, and General categories. With this, the FEG list was generated. Additionally, 114 students participated to determine the learning style according to the Bandler and Ginder instrument, focused on the Visual, Auditory and Kinesthetic model, generating the VAK list.

To measure the acceptance and efficiency of the personalized tutoring provided by the proposed architecture, a survey was carried out, adapted from the UMUX-Lite user experience and usability instrument, validated by an expert in pedagogical psychology. The

result of this study displayed a high level of acceptance in the personalized advice offered to students by the proposed software architecture and the integral environment. With the design and implementation of the proposed architecture, the results obtained, the detailed analysis of the results of the evaluations and a discussion of what was found, it was possible to determine the conclusions of the contributions of the study, lines of research feasible to explore and future work from the group of researchers participating in the study.

Keywords: Software architecture, Ontologies, xAPI, Intelligent Tutoring Systems

Dedicatoria

A dios por darme la oportunidad de realizar esta Doctorado.

A mis padres que siempre están ahí cuando los necesito, y por inculcarme la importancia de prepararme para la vida.

A mi esposa e hijos que son mi soporte y siempre me sostienen en cada paso que doy, por aguantar y apoyarme en todo lo que hago, por resistir los momentos difíciles que provoco cada cierre de semestre.

Agradecimientos

Quiero expresar mi agradecimiento:

Al Dr. Omar Álvarez Xochihua por ser mi director y guía, por siempre estar dispuesto a aportar y compartirme su conocimiento, por su paciencia y por el interés mostrado en este proyecto y contagiarme con entusiasmo para la realización de este proyecto.

Al Dr. José Ángel González Fraga por su apoyo y motivación para lograr este proyecto

A mi amigo Ing. Omar Cerecedo Zepeta por su apoyo incondicional, por siempre estar dispuesto a compartir su conocimiento.

A mis sinodales por compartir su tiempo y conocimiento, por su dedicación y paciencia para ayudarme a terminar mi trabajo de investigación.

A la exdirectora de la EEyG Ensenada, Norma Angélica Baylon Cisneros, por el apoyo recibido.

Al director de la FCAyS-Ensenada, Sergio Cruz Hernández, por su apoyo y disposición para ayudarme.

A mis compañeras Teresa Rousseau, Elia Pérez Nieto, Denisse Fragoso, Margarita González, Denisse Mejía por su apoyo incondicional.

Al personal administrativo de la Facultad de Ciencias y al personal del programa de Maestría y Doctorado en Ciencias e Ingeniería (MyDCI) por su atención, amabilidad y disposición para ayudarme.

Al Consejo Nacional de Ciencia y Tecnología (CONACyT) por el apoyo económico brindado, y a la Facultad de Ciencias de la Universidad Autónoma de Baja California (UABC) por las facilidades otorgadas para la realización de este trabajo.

A todos ustedes, muchas gracias.
Rodolfo Alan Martínez Rodríguez.

Contenido

Capítulo I.....	1
Introducción	1
1.1 Planteamiento del problema.....	4
1.2 Pregunta general.....	8
1.3 Preguntas específicas	9
1.4 Objetivos	9
1.4.1. Objetivo general	9
1.4.2. Objetivos Específicos.	10
1.5. Hipótesis	10
1.6 Justificación y delimitación de la investigación	10
1.6.1 Justificación de investigación	10
1.6.2 Limitaciones	13
1.6.2.1 Limitaciones de tiempo.....	13
1.6.2.2 Limitaciones de espacio y territorio.....	13
1.7 Estructura de la tesis	14
Capítulo 2.....	16
Revisión de la literatura	16
2.1. Evolución de los Sistemas de Tutoría Inteligente a nivel internacional	16
2.2. Líneas de investigación de los ITS	18
Marco Teórico.....	20
2.3. Sistemas de tutoría inteligente	20
2.4. Representación de conocimiento mediante Ontologías.....	23
2.4.1. Lenguajes ontológicos.....	24
2.4.2. RDF.....	24
2.4.3. Razonador.....	25
2.4.4. Modelos ontológicos orientados al estudiante	25
2.5 Comunicación multiagente	27
Capítulo 3.....	33
Metodología	33
3.1 Modelado de conocimiento y desarrollo tecnológico.....	34
3.1.1 Modelado de conocimiento basado en ontologías	34
3.1.2 Diseño de capas de la arquitectura integral de tutores inteligentes	39
3.2 Implementación del protocolo de comunicación entre tutores.....	40

3.2.1 Especificación xAPI	42
3.2.2 ¿Qué es un LRS (Learning Record Store)?	47
3.3 Evaluación del desempeño de la arquitectura de software propuesta al interactuar con múltiples tutores inteligentes provenientes de diversos dominios.....	49
3.4 Evaluación de la aceptación y eficiencia de la tutoría brindada	51
Capítulo 4.....	54
Diseño y desarrollo del ambiente integral de aprendizaje	54
4.1. Integración de xAPI en el proyecto RITADA	54
4.2. Evaluación preliminar del ambiente implementado.....	58
4.2.1 Participantes	58
4.2.2 Instrumentos	58
4.2.3 Procedimiento	59
4.2.4 Técnicas de análisis y evaluación preliminar	59
Capítulo 5.....	66
Evaluación de satisfacción y aceptación de uso de la asesoría brindada por la arquitectura propuesta	66
5.1 Participantes en el estudio de experiencia de usuario	66
5.2 Instrumentos utilizados en el estudio de experiencia de usuario	67
5.3 Procedimiento utilizado en el estudio de experiencia de usuario	69
5.4 Evaluación de aceptación de la arquitectura de software con base en el estilo de aprendizaje del estudiante.....	74
5.5 Evaluación de aceptación de la arquitectura de software con base en su nivel de autoestima.....	80
Capítulo 6.....	85
Conclusiones y trabajo futuro.....	85
6.1 Conclusiones:.....	85
6.2 Áreas de oportunidad y trabajo futuro.....	91
Referencias.....	94
Anexos	100
Anexo 1.....	100
Glosario de términos	100

Índice de figuras

Figura	Página
1.1. Índice de reprobación en al menos una unidad de aprendizaje en UABC.	6
1.2. Índice de reprobación en al menos una unidad de aprendizaje en UABC por ciudad.	7
1.3. Índice de reprobación en FCAyS UABC Ensenada en al menos una unidad de aprendizaje.	7
2.1. Proceso del ciclo de asesoría del ITS.	21
2.2. Ontología representando el modelo del estudiante.	26
2.3. Arquitectura de servicios de tutoría inteligente en un ambiente integrado.	29
3.1. Protege: ambiente para la creación de ontologías.	35
3.2. Nivel principal de la ontología representando el modelo del estudiante pedagógico.	35
3.3. Arquitectura multiagentes propuesta.	40
3.4. Evolución de especificación TIN CAN API (xAPI).	44
3.5. Estructura de declaraciones de LRS.	45
3.6. Estructura de declaraciones de LRS.	47
4.1. Arquitectura de servicios educativos mediante la especificación xAPI.	55
4.2. Arquitectura de servicios educativos mediante la especificación xAPI en la red federada RITADA.	57
4.3. Historial de registros derivado de la funcionalidad de un LRS.	57
4.4. Historial de acceso a los módulos de matemáticas y comunicación oral y escrita.	61
4.5. Historial de acceso a los módulos de hábitos de estudio y autoestima.	61
4.6. Imágenes del uso del sistema en forma concurrente.	62
4.7. Accesos concurrentes mediante sesión grupal día 1.	63
4.8. Accesos concurrentes mediante sesión grupal día 2.	63
4.9. Capas del modelo del estudiante propuesto en RITADA.	64
4.10. Implementación de H5P contenido multimedia interactivo para Web en RITADA	65

5.1. Ejercicio tipo de clasificación de palabras con base en su sílaba tónica en el tutor de Comunicación Oral y Escrita.	67
5.2. Ejercicio tipo de clasificación de palabras con base en su sílaba tónica utilizando texto e imágenes, en el tutor de Comunicación Oral y Escrita.	72
5.3. Ejercicio tipo de clasificación de palabras con base en su sílaba tónica utilizando sólo texto, en el tutor de Comunicación Oral y Escrita.	72
5.4. Ejercicio tipo de clasificación de palabras con base en su sílaba tónica, utilizando mensaje sutil “modifica tu respuesta” y sin imágenes, en el tutor de Comunicación Oral y Escrita.	73
5.5. Arquitectura de software RITADA.	73
5.6. Pregunta 1 por grupo y estilo de aprendizaje	76
5.7. Pregunta 2 por grupo y estilo de aprendizaje.	77
5.8. Pregunta 3 por grupo y estilo de aprendizaje.	77
5.9. Pregunta 4 por grupo y estilo de aprendizaje	79
5.10. Pregunta 5 por grupo y estilo de aprendizaje	79
5.11. Pregunta 6 por grupo y estilo de aprendizaje	80
5.12. Pregunta 1 por grupo y nivel de autoestima	82
5.13. Pregunta 2 por grupo y nivel de autoestima.	82
5.14. Pregunta 7 por grupo y nivel de autoestima.	83
5.15. Pregunta 8 por grupo y nivel de autoestima.	84

Índice de tablas

<u>Tabla</u>	<u>Página</u>
1.1. Lista de tutores inteligentes a nivel internacional.	3
5.1. Feedback-Estilo de aprendizaje-Autoestima.	
5.2. Clasificación de estudiantes participantes en el estudio, con base en su estilo de aprendizaje predominante.	
5.3. Clasificación de estudiantes participantes en el estudio, con base en su nivel de autoestima.	

Capítulo I

Introducción

Las tecnologías de la información y la comunicación (TIC) están en constante evolución, impulsando el desarrollo económico, social y cultural alrededor del mundo [1]. Organizaciones pertenecientes a los tres sectores económicos que definen el progreso de un país, primario (extracción de materias primas), secundario (manufactura) y terciario (servicios), han sido sustancialmente beneficiadas con el uso de esta tecnología [2]. Específicamente, en el tercer sector, el cual agrupa empresas e instituciones que ofrecen servicios, encontramos un gran avance tecnológico, tanto en sus procesos administrativos como operativos. Por ejemplo, empresas que brindan servicios de seguridad, salud, recreación, educación, entre otras, hacen uso de las TIC para hacer más eficientes sus servicios, reducir costos de operación e innovar.

Dentro del tercer sector, las instituciones educativas hacen un amplio uso de las TIC en el proceso de enseñanza-aprendizaje. Sin embargo, esta tecnología ha sido principalmente utilizada para replicar las actividades de enseñanza convencionales. Siendo el avance de la misma tecnología, y el uso que los distintos grupos sociales hacen de ésta, los que demandan un uso más sofisticado de la misma para atender funciones específicas. Particularmente, en el ámbito del acompañamiento escolar, los estudiantes que reciben educación presencial demandan atención y asesoría del profesor lo más actualizada y personalizada posible, en ocasiones imposibles de brindar. En ambientes de aprendizaje parcial o totalmente en línea, que son cada vez más utilizados, los estudiantes generalmente recurren a interacciones asíncronas (ej. foros de discusión) para solicitar asesoría de otros usuarios con experiencia en el dominio de interés, generando en ocasiones un detrimento en la calidad y fluidez en la obtención del conocimiento.

Debido a que, en la enseñanza bajo el esquema presencial, tanto en escuelas públicas como privadas, es imposible brindar atención personalizada a cada estudiante, así como la importancia de brindar asesoría efectiva y en tiempo real a estudiantes bajo un entorno en línea, se ha buscado hacer uso de tecnología para brindar asesoría de calidad a nuestros estudiantes mientras tratan de obtener nuevo conocimiento. En este ámbito,

existen grupos a nivel internacional atendiendo diversas áreas de investigación educativa, tales como: *Apoyo inteligente para el aprendizaje*, *Minería de datos educativa y análisis del aprendizaje*. Específicamente, los sistemas de tutoría inteligente (ITS, por sus siglas en inglés), asesores virtuales basados en software, surgen como una opción para incrementar el desempeño académico de los estudiantes al brindarles asesoría personalizada; mediante conocimiento previamente obtenido de asesores humanos. En la actualidad, podemos aseverar que este tipo de sistemas han sido implementados con éxito atendiendo una diversidad de dominios (ej. matemáticas, idiomas y medicina), cubriendo todos los niveles educativos.

Los ITS han sido investigados durante varias décadas , reportándose un progreso significativo en el área gracias a los avances tecnológicos de comunicación (internet) y de representación de conocimiento (inteligencia artificial). Como resultado de este avance, la aplicación de sistemas de aprendizaje basados en tutores inteligentes recibe cada vez más atención en el sector educativo. Lo anterior ha permitido desarrollar ITS en una diversidad de dominios, tales como: el Cognitive Tutor, desarrollado en Estados Unidos, que atiende temas de matemáticas para diversos niveles educativos [3]; y el ITS para la nivelación académica universitaria, desarrollado a nivel nacional, orientado a reforzar competencias en los dominios de matemáticas preuniversitarias y comunicación oral y escrita [4]. En la Tabla 1.1 se presenta una lista más completa en orden cronológico de los ITS que han sido más destacados en la literatura.

Sin embargo, el desarrollo de este tipo de sistemas se ha dado de forma aislada, resaltando la inexistencia de una arquitectura de software que permita la comunicación e interacción entre tutores inteligentes, acción que les permitiría compartir y cooperar al realizar su labor de asesorar a sus usuarios.

En este trabajo de investigación doctoral se describe el diseño y desarrollo de una arquitectura de software que, mediante la implementación de protocolos de comunicación y cooperación inter-tutores, permita utilizar un modelo del estudiante compartido entre distintos tutores inteligentes; modelo que representa el nivel cognitivo y conductual de cada alumno y que contiene un alto nivel de similitud entre un tutor y otro. En la actualidad, el empleo de la inteligencia artificial en el ámbito educativo es una herramienta que permite potenciar el desempeño escolar de los estudiantes. En específico,

mediante la aportación propuesta de un modelado de conocimiento compartido que se sustente en el área de estudio de tutores inteligentes.

Tabla 1.1. Lista de tutores inteligentes a nivel internacional (elaboración propia).

NOMBRE	DOMINIO	SECTOR	AÑO
SRLSS	Matemáticas	Educación	2018
ITSS	Lectura	Educación	2017
SITS	Redes bayesianas	Educación	2016
Inq-ITS	Psicología	Educación	2015
ITS matemáticas básicas	Matemáticas	Educación	2014
Crystal Island	Ciencia	Educación	2013
Zosmat	Psicología	Educación	2009
Wayang	Matemáticas	Educación	2009
eTeacher	Redes valencianas	Educación	2008
Virtual Reality Exposure	Medicina	Medicina	2005
Andes	Física	Educación	2005
battlefield Command	Estrategias de combate	Militar	2002
Simulation Patient Cardiac	Medicina	Medicina	1994
Sherlok	Electrónica	Fuerza Aérea	1992
Helicopter Pilot USArmy	Aviación	Militar	1990
Auto tutor	Informática	Educación	1990
Andes	Física	Militar	1988
Sophie	Electrónica	Educación	1973
Scholar	Ciencias Naturales	Educación	1970
Meno	Computación (programación)	Educación	1970

El modelado de conocimiento de tutores inteligentes, la implementación de la arquitectura de software que ofrezca un ecosistema de aprendizaje y la oportunidad de potenciar la personalización de la asesoría brindada mediante el apoyo de estos sistemas, denotan la relevancia de esta tesis centrada en el estudio de tutores inteligentes.

El tema de investigación fue delimitado por diversos factores, tales como el resultado de la revisión de literatura en las áreas de estudio de tutores inteligentes y ecosistemas de aprendizaje, la experiencia previa en el uso y desarrollo de tutores inteligentes por parte del grupo de investigación, y la necesidad de mejorar la personalización de la asesoría brindada por este tipo de sistemas.

1.1 Planteamiento del problema

La deserción escolar, los bajos índices de aprobación en las unidades de aprendizaje y el rezago escolar son los problemas a los que se enfrentan las instituciones educativas de educación superior, problemas que son reconocidos por las mismas instituciones educativas [5].

En el trabajo presentado en [6] mencionan que uno de los retos principales de la educación superior es disminuir la deserción e incrementar la retención estudiantil y la eficiencia terminal. De acuerdo al estudio presentado en [7], sobre los indicadores del sistema educativo en México durante el ciclo escolar 2016-2017, muestran un porcentaje de absorción nacional a nivel superior, es decir los alumnos que finalizan un nivel educativo es del 73% y un abandono escolar del 7.2%. Mientras que para el periodo 2017-2018, se reporta un 74% y 6.8%, respectivamente; indicando que, de 3,864,995 alumnos pertenecientes al nivel superior, en la modalidad escolarizada, 262,820 han abandonado la escuela. Existen diversos factores que inciden en la deserción estudiantil. En [8], se proponen 14 factores que detonan la reprobación y deserción, dividiéndolos en dos categorías, una atribuible al alumno y otra atribuible al profesor, entre estos factores destaca la dinámica pedagógica y la relación alumno-maestro.

Por otra parte, en [9], sin dejar de lado los factores sociales, hace alusión a problemas atribuibles tanto a la institución como el maestro. Por otro lado, en [10] se consideran factores relacionados con los maestros, el alumno y el tiempo dedicado al estudio. Finalmente, en [11] hacen la propuesta de nueve factores, entre los que destaca la ausencia de asesoría grupal o personalizada, y el énfasis en la relación alumno-profesor.

En ese sentido, se busca destacar que la relación entre docente y alumno, desde la perspectiva de la posibilidad para acercarse al maestro en caso de tener dificultades académicas, así como la habilidad del docente para adaptarse a las necesidades de aprendizaje específicas del alumno, son factores determinantes para la permanencia del estudiante y la conclusión de sus estudios de manera eficiente. Sin embargo, una matrícula elevada, reduce la interacción con el alumno y el tiempo de atención efectivo para subsanar deficiencias.

Por otro lado, de acuerdo con la investigación publicada en [12], se prevé un crecimiento en las matrículas escolares, lo anterior producto de la reforma educativa

publicada en el Diario Oficial de la Federación el 15 de Mayo de 2019, donde se establece una modificación al artículo 3ro de la Constitución Política de los Estados Unidos Mexicanos, el cual establece que toda persona tiene derecho a la educación. Donde el Estado y la Federación deberán impartir y garantizar la educación inicial (preescolar, primaria y secundaria), así como que la educación media superior y superior serán obligatorias. Adicionalmente, la nueva reforma educativa no contempla la realización de evaluación a los docentes, es decir no se determinará la permanencia de los docentes en sus puestos por los resultados que obtengan en las evaluaciones a las que sean sometidos.

En este mismo sentido, la iniciativa de reforma educativa que presentó el Ejecutivo Federal en diciembre del 2019, plantea nuevas presiones a las universidades, ya que en su actual redacción se establece como obligatoria y gratuita la educación superior, que abarca nivel el técnico superior, la licenciatura y los posgrados. Por lo cual, se espera un aumento significativo en la matrícula en las instituciones de educación superior, que actualmente se estima en un 38% [13]. Así, pues, se plantea un incremento estudiantil, pero no se visualiza un crecimiento en la planta docente acorde al número de alumnos matriculados en cada institución educativa.

Por otro lado, una de las mayores dificultades que confrontan los sistemas educativos latinoamericanos, es el bajo rendimiento académico de los estudiantes, definido éste como el nivel de aprendizaje alcanzado por los alumnos durante el proceso instruccional [14]. Otro aspecto que genera dificultades es la carencia de una atención individualizada, la aplicación de estrategias didácticas apropiadas, la poca formación del docente y la motivación no adecuada de los jóvenes por continuar los estudios [15].

Para remediar este problema a nivel nacional, la Asociación de Universidades e Instituciones de Educación Superior (ANUIES) desarrolló en el año 2000 un documento titulado *La Educación Superior en el Siglo XXI*. En esta publicación se establece un programa de educación integral para los alumnos, con la finalidad de apoyar al sistema de educación superior (SES) a través de programas de tutoría para apoyar al alumnado y lograr que una mayor parte de los estudiantes inscritos en los programas educativos logren culminar sus estudios y alcanzar sus objetivos [12].

De igual modo, otra manera para tratar de subsanar el incremento en la matrícula y la carga hacia los docentes es a través de la educación a distancia. Sin embargo, en el

sector educativo nacional se encuentra con una gran resistencia hacia las modalidades semipresenciales y a distancia. Particularmente, en la Universidad Autónoma de Baja California (UABC) los índices de reprobación generales de la institución, durante el primer periodo del año 2015, alcanzaron un nivel donde se reporta que el 47% de sus estudiantes reprueba al menos una materia. Para el segundo periodo de ese mismo año, se reportó un leve descenso con un índice de reprobación del 46%. Mientras que para el ciclo 2016, en sus dos periodos, el comportamiento a nivel estado incrementó hasta el 50%. En el 2017, en su primer semestre, se reportó un 48% de índice de reprobación y para el segundo semestre un descenso al 44%. En el 2018 se reportó para el primer periodo un incremento al 48% y para el segundo periodo al 51% (ver Figura 1.1). En términos generales se puede decir que es poca la variación entre ciclos y que es prácticamente la mitad del alumnado quien no obtiene calificación aprobatoria en al menos una materia.

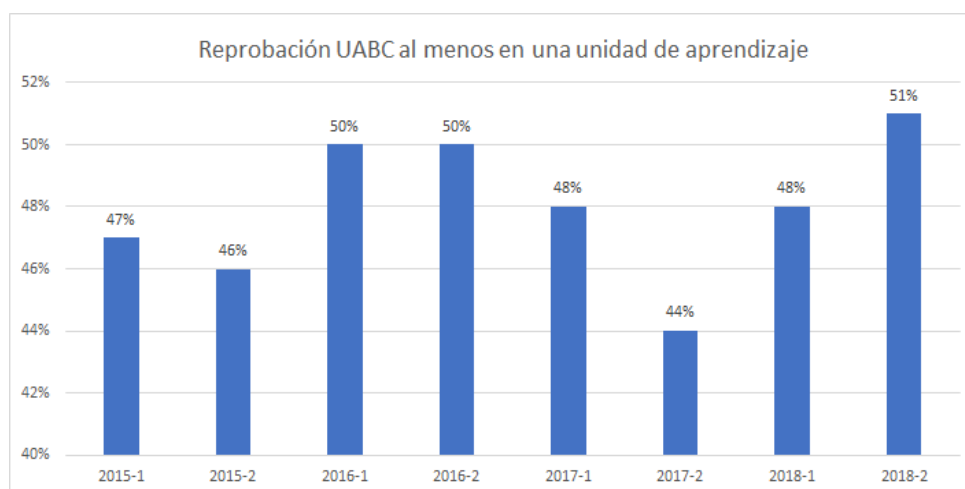


Figura 1.1. Índice de reprobación en al menos una unidad de aprendizaje en UABC
Fuente: elaboración propia con datos tomados del sistema de indicadores UABC (2019)

Este patrón de reprobación no se encuentra diferenciado significativamente por municipio o programa educativo, sino que es una tendencia generalizada (ver Figura 1.2), oscilando entre el 39% y 54%. Siendo Tijuana el municipio que reporta un índice de reprobación más bajo, de 39% al 49%, y Mexicali el más alto, del 48% al 54%. Un promedio de los ocho períodos registrados (cuatro años), nos da un índice de reprobación del 48.38% para Ensenada, de 44.75% para Tijuana y de 51.38% para Mexicali, valores por demás elevados y similares, lo que nos indica que la reprobación no está focalizada, sino que es general e indistinta con respecto al programa educativo y campus.

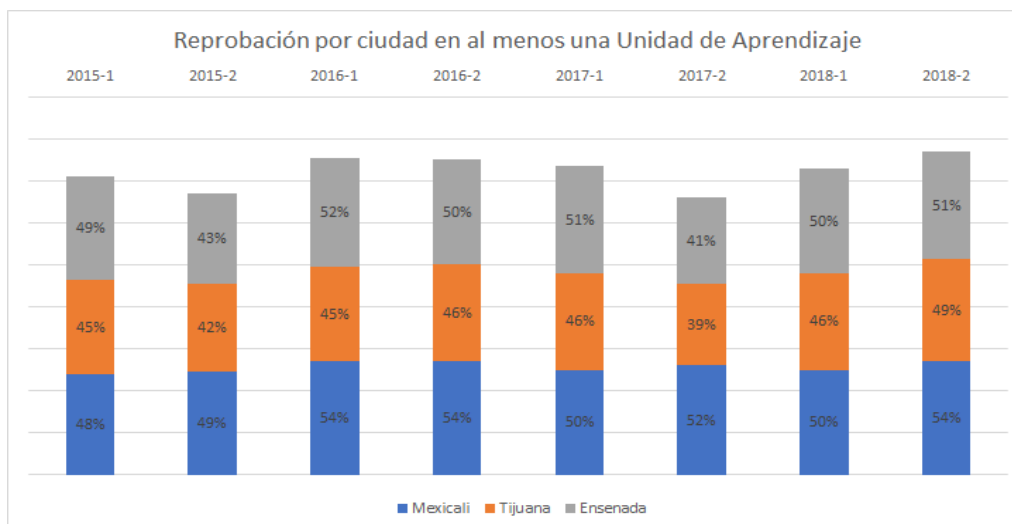


Figura 1.2. Índice de reprobación en al menos una unidad de aprendizaje en UABC por ciudad
Fuente: elaboración propia con datos tomados del sistema de indicadores UABC (2019).

De manera más particular, la Facultad de Ciencias Administrativas y Sociales (FCAyS), no presenta un patrón diferente a la tendencia general de la universidad, con un nivel de reprobación que va del 38% al 45%, que en promedio tiene un valor de 42.63% (ver Figura 1.3)

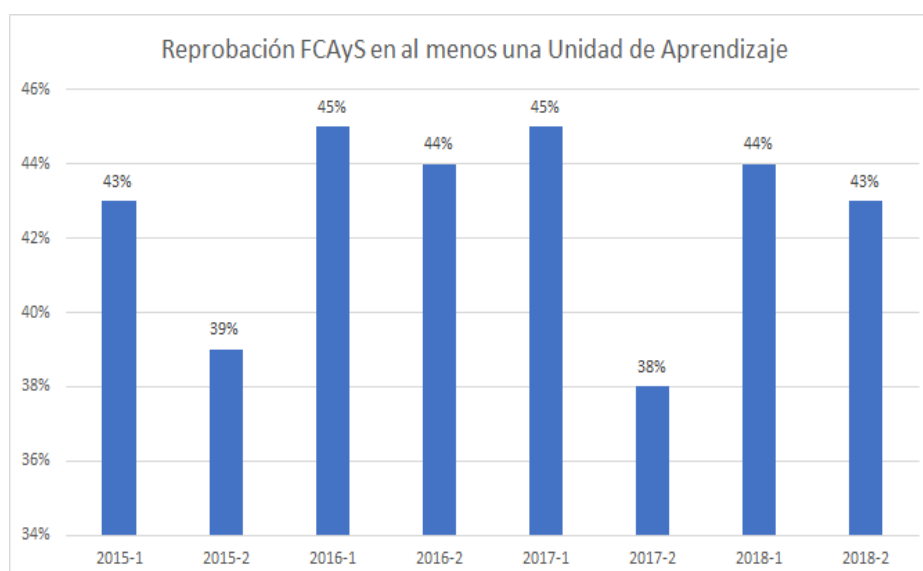


Figura 1.3. Índice de reprobación en FCAyS UABC Ensenada en al menos una unidad de aprendizaje.

Fuente: elaboración propia con datos tomados del sistema de indicadores UABC (2019).

Ante estas evidencias, respecto a los altos índices de reprobación, al incremento en la matrícula que se prevé, y la poca aceptación en las escuelas de las plataformas actuales para asesorías y cursos de manera no presencial, es que se identifica una

problemática importante. Es imprescindible contar con un mayor apoyo docente al estudiante a través de brindar una asesoría personalizada, misma que sea rápida, efectiva y confiable; la cual dada la matrícula actual y la proporción alumno-docente, que es de alrededor de un maestro por cada 40 estudiantes, resulta ineficiente que se realice exclusivamente de manera presencial y personalizada. Con base en lo anterior, se determina que el apoyo tecnológico es requerido y urgente. Siendo así que la arquitectura de software propuesta en este trabajo, permitiendo la comunicación entre tutores inteligentes, podría facilitar dicho proceso de tutoría. Trayendo beneficios al ofrecer asesoría personalizada al compartir los perfiles de aprendizaje de cada estudiante conformados por sus hábitos de estudio, autoestima y estilo de aprendizaje.

El diseño de esta arquitectura propuesta, resulta pertinente y oportuna para subsanar las deficiencias en el modelo tradicional que implementa la UABC y otras instituciones educativas, pudiendo este, replicarse en otras universidades. El impacto de este trabajo, pretende beneficiar directamente a los más de 4500 estudiantes actualmente matriculados en FCAYS reportados en el último informe directivo 2017-2018, a los que se prevé ingresen en próximos ciclos, a los docentes e idealmente, en un futuro, a otras unidades académicas e instituciones [16].

El objetivo de la presente tesis consiste en desarrollar y evaluar el funcionamiento de una arquitectura basada en software, robusta y escalable, que soporte un ambiente interactivo de tutores inteligentes, permitiendo la cooperación y coordinación de las actividades de asesoría, así como homogeneizar y compartir la base de conocimiento del estudiante lo cual nos lleva a plantearnos las siguientes preguntas:

1.2 Pregunta general

¿Cómo implementar y evaluar el funcionamiento de una arquitectura basada en software que soporte un ambiente interactivo de tutores inteligentes, permitiendo la cooperación y coordinación de las actividades de asesoría, así como estandarizar y compartir la base de conocimiento del estudiante?

1.3 Preguntas específicas

1. ¿Qué esquema de representación de conocimiento permite la implementación del modelo del estudiante que facilite su integración con modelos de experto independientes del dominio?
2. ¿Qué modelo de arquitectura de software permitiría la cooperación e interacción entre múltiples tutores inteligentes, al compartir su modelo del estudiante?
3. ¿Qué protocolo de comunicación entre tutores inteligentes, basado en la tecnología multiagentes, permite la interacción entre ellos?
4. ¿Cómo lograr un desempeño operativo de la arquitectura propuesta al interactuar con múltiples tutores inteligentes de diversos dominios, considerando sus características de operatividad, rendimiento, compatibilidad, y escalabilidad?
5. ¿Qué nivel de aceptación y eficiencia se logra al brindar asesoría, mediante un ambiente interactivo de tutores inteligentes que estandariza y comparte la base de conocimiento del estudiante?

1.4 Objetivos

Derivado de las preguntas de investigación se establece el objetivo General y los objetivos específicos:

1.4.1. Objetivo general

Desarrollar y Evaluar la funcionalidad de una arquitectura basada en software, robusta y escalable, que soporte un ambiente interactivo de tutores inteligentes, permitiendo la cooperación y coordinación de las actividades de asesoría, así como homogeneizar y compartir la base de conocimiento del estudiante.

1.4.2. Objetivos Específicos.

1. Establecer un esquema de representación de conocimiento para la implementación del modelo del estudiante que permita su integración con modelos de experto independientes del dominio.
2. Crear una arquitectura de software que permita la cooperación e interacción entre múltiples tutores inteligentes, al compartir su modelo del estudiante.
3. Establecer un protocolo de comunicación entre tutores inteligentes, basado en la tecnología multiagentes.
4. Lograr un desempeño operativo de la arquitectura propuesta al interactuar con múltiples tutores inteligentes de diversos dominios, considerando sus características de operatividad, rendimiento, compatibilidad, y escalabilidad
5. Evaluar la aceptación y eficiencia de la tutoría brindada a estudiantes mediante la arquitectura de tutores inteligentes propuesta.

Con base en las preguntas de investigación, consideramos las siguientes hipótesis:

1.5. Hipótesis

H₁. Mediante el uso del ambiente integral de tutores inteligentes propuesto, se impactará positivamente en la integración de tutores inteligentes mediante el uso de un modelo del estudiante compartido y en la calidad de la asesoría brindada.

1.6 Justificación y delimitación de la investigación

1.6.1 Justificación de investigación

Desde una perspectiva práctica, la arquitectura de software propuesta busca subsanar las deficiencias trascendentales para una tutoría virtual eficiente. En primer lugar, pese a que la tecnología ha avanzado sustancialmente en las últimas décadas, el

tiempo y costo que consume el desarrollar un ITS es similar al invertido cuando surgieron los tutores inteligentes, en [17] se reporta que el tiempo promedio de desarrollo de un ITS es de alrededor de 200-300 horas por cada hora de sesión de clase que el ITS brinde, siendo el tiempo de desarrollo invertido significativamente mayor al de la asesoría que el ITS proporcionará; dependiendo el dominio para el que se está diseñando. El mayor tiempo y atención se invierte en el diseño y materialización del modelo del dominio, restándole importancia a los modelos del estudiante y pedagógico. No obstante, estos dos últimos modelos son considerados vitales para garantizar el adecuado funcionamiento de un tutor inteligente, siendo estos subestimados al asignarles recursos para su implementación.

El proceso tradicional para el desarrollo de un ITS está compuesto de tres módulos, el primero es el módulo que implementa el modelo del dominio, que representa el conocimiento de expertos. Generalmente, mediante estructuras de representación de conocimiento, se modela el cómo solucionar problemas de dominios específicos, el conjunto de respuestas correctas para cada problema, errores más comunes cometidos por el estudiante y la asesoría brindada cuando esta es llevada a cabo.

El segundo módulo representa al modelo del estudiante, el cual modela el nivel de conocimiento de cada uno de los usuarios del sistema (alumnos). Típicamente incluye errores comunes del estudiante, tiempo requerido e intentos para solucionar un problema específico, respuestas correctas e incorrectas, así como información alterna, o no-cognitiva, como su estilo de aprendizaje y hábitos de estudio.

El tercero es el modelo pedagógico que es el responsable de establecer el tipo y forma de brindar una asesoría. Lo anterior considerando el perfil cognitivo y estilo de aprendizaje de cada estudiante y mediante la interpretación de sus competencias y deficiencias en un área de conocimiento objetivo. Formando una imagen instantánea y adaptativa del nivel personalizado de comprensión de contenidos, este módulo constituye una característica muy importante para los ITS, que es su capacidad de adaptarse a las necesidades de cada estudiante.

Si bien, al momento no es posible cuantificar de manera precisa la reducción del tiempo que la arquitectura puede permitir, ni es este el objetivo principal de la investigación. El hecho de contar con un modelo del estudiante compartido, permitirá que

este no requiera rediseñarse en cada desarrollo de un nuevo ITS, sino que este sirva como patrón o modelo con capacidad de ser replicado cuantas veces sea necesario.

Por otro lado, si bien el uso de tutores inteligentes ha sido ampliamente referido en el ámbito educativo, en sus diferentes niveles, tanto a nivel nacional como internacional, como una herramienta que busca la mejora en el desempeño escolar [18], siguen siendo considerados como un área de investigación abierta. El uso de esta tecnología enfrenta el problema que, más allá del tiempo y los recursos de desarrollo [17], existe el aislamiento o incomunicación entre ITS que atienden dominios distintos, desaprovechando la compartición de perfiles educativos de estudiantes comunes, que permita elaborar mejores estrategias de enseñanza. Mediante la arquitectura propuesta, es posible entonces resarcir esa deficiencia a través de homogeneizar el modelo del estudiante, permitiendo dar una asesoría personalizada acorde a las necesidades de cada usuario.

Adicionalmente, en la revisión de literatura no se ha considerado la cooperación e interacción entre ITS en beneficio de la calidad de la asesoría ofrecida. En esta investigación se asume que la estandarización en el modelado del conocimiento del estudiante, a través de la cooperación entre tutores inteligentes, permitirá la interacción y colaboración entre ITS que atienden dominios distintos e incrementar el impacto de los mismos.

Siendo el resultado de esta investigación de relevancia para el área de estudio *Apoyo Inteligente para el Aprendizaje*, y en particular para los *Sistemas de Tutoría Inteligente*, es importante precisar las siguientes contribuciones: 1) técnicas de modelado del conocimiento del estudiante, que faciliten la cooperación e intercambio de información entre tutores, y 2) una arquitectura robusta y escalable que soporte la integración de funcionalidades y la base de conocimiento de múltiples ITS, a través de protocolos de comunicación y estándares de interacción. En segundo término, aunque se considera la existencia de un alto grado de generalización al modelar los aspectos cognitivos y conductuales del estudiante, así como de las estrategias didácticas y pedagógicas que soportan la actividad de asesoría, no existen estándares o especificaciones técnicas que nos permitan implementar estos modelos de forma eficiente y efectiva.

1.6.2 Limitaciones

Por otra parte, pese a que el proyecto se puede desarrollar en los diferentes niveles educativos, la parte evaluativa de la investigación se llevó a cabo con estudiantes del nivel educativo superior y medio superior.

1.6.2.1 Limitaciones de tiempo

Debido a las restricciones en el tiempo, en este trabajo fue posible medir la aceptación y comodidad de los usuarios con respecto a la arquitectura. Para que pudiese ser valorada la eficiencia de la misma, sería necesaria la realización de un estudio longitudinal donde pudiesen compararse los resultados derivados de la tutoría tradicional versus la de la arquitectura propuesta en este trabajo.

1.6.2.2 Limitaciones de espacio y territorio

Si bien el proyecto se puede replicar en instituciones de cualquier zona geográfica, las pruebas de la arquitectura fueron destinadas a estudiantes de la Universidad Autónoma de Baja California campus Valle Dorado, Ensenada Baja California.

La presente investigación se centra en fomentar el uso de la tecnología de la información en el ámbito educativo. Siendo el interés principal potenciar el desempeño escolar de estudiantes universitarios utilizando herramientas de inteligencia artificial, particularmente mediante la aportación en el modelado de conocimiento compartido, sustentado en el área de estudio de tutores inteligentes.

El tema de investigación está delimitado por diversos factores, tales como el resultado de la revisión de literatura en las áreas de estudio de tutores inteligentes y ecosistemas de aprendizaje, la experiencia previa en el uso y desarrollo de tutores inteligentes, y la necesidad de mejorar la personalización de la asesoría brindada por este tipo de sistemas.

1.7 Estructura de la tesis

La presente tesis se encuentra dividida en seis capítulos, en el primer capítulo se presenta la naturaleza y dimensión del tema de investigación, para ello se presenta el planteamiento del problema, la justificación, el objetivo, tanto general como los específicos, preguntas de investigación y las limitaciones existentes para llevar a cabo este estudio.

El segundo capítulo incluye la revisión de la literatura organizada en tres temas fundamentales para esta investigación: sistemas de tutoría inteligente, representación del conocimiento del estudiante mediante ontologías, y agentes inteligentes y sistemas multiagentes. El análisis de literatura presentado en el capítulo tiene por objetivo el aportar una revisión detallada de investigaciones sobre los avances que se han reportado en esta área de estudio, así como presentar una revisión de estudios previos de los temas incluidos en esta investigación.

En la investigación del primer tema, sistemas de tutoría inteligente, se describe a los tutores inteligentes y su evolución en el área de inteligencia artificial, seguido de un análisis de estudios previos de implementaciones de este tipo de sistemas a nivel internacional, finalmente se mencionan las líneas de investigación que se han abordado en el estudio de tutores inteligentes.

En el desarrollo del segundo tema de investigación, representación del conocimiento del estudiante mediante ontologías, se describe a las ontologías desde el origen de su terminología filosófica hasta el término en el área de ciencias computacionales, sus componentes y lenguajes de desarrollo, hasta llegar al estudio de modelos ontológicos.

En el tercer tema de investigación, agentes inteligentes y sistemas multiagentes, se aborda desde la descripción de agentes inteligentes y sistemas multiagentes, seguido del análisis de las arquitecturas de sistemas multiagentes, así como los estudios previos reportados en su implementación. Haciendo énfasis en los ecosistemas digitales de aprendizaje ya que son parte fundamental en esta investigación.

En el tercer capítulo se detalla la metodología utilizada en la presente investigación para el modelado del conocimiento y desarrollo tecnológico, seguido del proceso utilizado para la investigación evaluativa del sistema propuesto.

En el cuarto capítulo se describe el diseño y desarrollo del ambiente propuesto, producto de esta investigación, Sistema Integral de Tutoría Inteligente (SITI). Se incluye una descripción sobre las áreas de conocimiento afines a lo implementado, el diseño de interfaces y la implementación del ambiente.

Por otra parte, en el quinto capítulo se describe el mecanismo de evaluación utilizado para medir el funcionamiento de la arquitectura de tutores inteligentes resultante, el nivel de aceptación de la asesoría brindada por el ambiente, y se discuten los resultados obtenidos en dos etapas de evaluación del SITI. En el sexto capítulo se finaliza con las conclusiones obtenidas durante el proceso del diseño, desarrollo y evaluaciones del ambiente propuesto, adicionalmente se presenta el trabajo futuro.

El estudio se llevó a cabo con la colaboración de la Facultad de Ciencias Administrativas y Sociales (FCAyS). El apoyo consistió en permitir el uso del SITI a estudiantes de múltiples grupos del tronco común del área administrativa y social en los ciclos de 2017 a 2019.

La aportación principal que ofrece esta investigación al área de estudio de ITS, es el diseño e implementación de un modelo del estudiante generalizado y una arquitectura de software que permite la cooperación y la optimización operativa de múltiples tutores inteligentes al compartir la base de conocimiento del estudiante y estrategias pedagógicas.

Capítulo 2

Revisión de la literatura

2.1. Evolución de los Sistemas de Tutoría Inteligente a nivel internacional

Hasta mediados de los años 60 se llegó a pensar que en breve las computadoras lograrían simular la inteligencia humana, pero a través del proceso evolutivo de esta tecnología se ha aprendido que con tan solo perfeccionar los componentes físicos (hardware) no era suficiente. Investigadores en el área de Inteligencia Artificial, al tratar de imitar la inteligencia humana, se encontraron con un gran reto al intentar definir formas efectivas de *representar, procesar y transmitir* el conocimiento humano [18].

En el ámbito educativo, se ha buscado generar sistemas inteligentes que repliquen el proceso de enseñanza aprendizaje. Particularmente, a través del desarrollo de Sistemas de Tutoría inteligente, se busca representar el conocimiento adquirido por profesores o expertos en algún dominio específico, para posteriormente ser transmitido de forma efectiva a nuevos estudiantes [19].

Las primeras investigaciones sobre ITS se enfocaron en definir la arquitectura que debe soportar un tutor inteligente, concluyendo que estos no solo deben representar conocimiento del dominio. Adicionalmente, se determinó la importancia de representar el nivel cognitivo de los estudiantes y las estrategias y competencias necesarias para transmitir el conocimiento. Dichas representaciones son referidas en el área de tutores inteligentes como *modelo de dominio, modelo del estudiante y modelo pedagógico*, respectivamente [20].

En una segunda etapa, se empezaron a desarrollar tutores inteligentes en diversos dominios, con el fin de evaluar su impacto real en el desempeño académico de los estudiantes que los utilizaban. A finales de los 70 e inicios de los 80, investigadores de las propias instituciones educativas eran los que realizaban estudios e implementaciones de ITS y los aplicaban con sus mismos estudiantes. Proyectos como *Scholar* [21], *Meno*

[22] y *Sophie* [23], son ejemplos de tutores inteligentes desarrollados para el dominio de temáticas en el área de electrónica.

Los ITS comenzaron a desarrollarse bajo este nombre en la década de los 80, con la idea fundamental de impartir conocimiento guiando al estudiante en el proceso de aprendizaje mediante la implementación de alguna forma de inteligencia [18]; considerando que el conocimiento se debería de adaptar al nivel cognitivo y conductual del estudiante. En esta década surgen proyectos tales como *Andes* [24], donde la milicia de Estados Unidos empieza a intervenir e invertir en esta tecnología para aplicarla en cursos de Física para militares.

Ya en los años 90, como resultado del impacto que tuvieron los tutores inteligentes en el adiestramiento militar, el gobierno de Estados Unidos invierte en el desarrollo de ITS para los dominios de medicina, psicología cognitiva, fuerza aérea y marina. Dos de los proyectos que destacaron en este periodo fueron “*the simulation patient cardiac*” [25] y “*sherlock*” [26]; que utilizando las nuevas tecnologías mostraron un gran paso en la elaboración y desarrollo de ITS con un enfoque de asesoramiento más definido.

En la década del 2000 se empieza a considerar al tutor humano tradicional como una alternativa de enseñanza continua para los tutores inteligentes, generando ITS adaptables a nuevos conocimientos y a la capacidad de evolución personalizada para cada estudiante. Es en esta década cuando se empiezan a desarrollar más tutores inteligentes atendiendo una diversidad de áreas de estudio (ej. matemáticas, idiomas, programación de computadoras, física, entre otros), y el uso de ellos llega a las instituciones de todos los niveles educativos.

En esta década se han desarrollado ITS que están siendo utilizados a gran escala, tales como el *cognitive tutor Álgebra 1* dentro del dominio de las matemáticas [27]; tutor de álgebra dirigido a estudiantes de primer año de secundaria. Álgebra 1 está diseñado para alumnos con diferentes habilidades y nivel de conocimiento en el área. Este sistema incluye materiales de texto y ofrece tutorías automatizadas dirigidas a preparar a los alumnos de primer año con temas básicos antes de pasar a temas más avanzados. En la actualidad, el surgimiento de Internet y los nuevos lenguajes de programación, ha permitido que los proyectos de ITS mejoren en el manejo de contenidos, accesibilidad y usabilidad para el estudiante.

A nivel nacional contamos con proyectos como el desarrollado en la Universidad Autónoma de Baja California por la *Red para la Innovación Tecnológica en Ambientes Digitales de Aprendizaje* (RITADA), en el cual se cuenta con servicios de contenido educativo y ejercicios con asesoría inteligente en las áreas de matemáticas preuniversitarias, comunicación oral y escrita, hábitos de estudio y autoestima [28]. Por otra parte, en el Instituto Tecnológico de Culiacán se llevó a cabo el proyecto de un sistema de tutoría inteligente para la enseñanza de matemáticas a estudiantes de tercer grado de primaria para identificar el estado emocional del estudiante y dar retroalimentación en sus cursos de matemáticas con base en expresiones faciales [29].

2.2. Líneas de investigación de los ITS

Actualmente, dentro del campo de estudio de tutores inteligentes, los diferentes grupos de investigadores han enfocado sus estudios en áreas como:

- 1) Perfeccionamiento del modelado de conocimiento del experto y del estudiante.
- 2) Uso de sensores para adquirir y representar aspectos emocionales y contextuales de los estudiantes.
- 3) Evaluación del impacto real que se logra en el desempeño del estudiante.
- 4) Evaluación del nivel de satisfacción o experiencia de usuario del estudiante.
- 5) Predicción del desempeño del estudiante.

Con relación al uso de inteligencia artificial y analítica de aprendizaje, en la literatura encontramos diversos trabajos de investigación aplicada [25,26,30,31] donde se utiliza una variedad de técnicas de AI. Por ejemplo, mediante el modelado de conocimiento utilizando redes neuronales se ha logrado predecir las calificaciones de un alumno con base en sus resultados en ejercicios previos. Las redes neuronales son elementos simples interconectados masivamente en paralelo, que interactúan entre sí, tal como lo haría un sistema neuronal humano [25]. Similarmente, en la investigación presentada en [26], los autores proponen que el módulo del tutor disponga de un listado

de protocolos pedagógicos o métodos de enseñanza que permita decidir cuál es la estrategia didáctica más adecuada con base en los datos de evaluaciones parciales y finales del estudiante. Adicionalmente, a partir del análisis de los datos generados al interactuar con el contenido instruccional, se busca efectuar un diagnóstico con base en los errores cometidos. De este modo, es posible predecir errores y dar sugerencias para corregirlos. Nuevamente, para la predicción se utiliza una red neuronal que le sugiere al estudiante una serie de ejercicios personalizados con el fin de mejorar su desempeño escolar. Otros modelados de conocimiento se realizan mediante el uso de redes bayesianas, conocidas como redes causales o probabilísticas, presentando un modelo que permite representar la incertidumbre del estudiante y su nivel de conocimiento [30].

En el área de la computación afectiva, en la que se debe lograr que las computadoras comprendan emociones del ser humano, existen investigaciones recientes que han demostrado que las emociones pueden influir en el comportamiento humano y en sus habilidades de aprendizaje. En el estudio realizado en [31], se presenta un análisis de los métodos de reconocimiento de emociones para adaptarlos a la tutoría, en el que se analizan los tutores afectivos *Auto Tutor*, *Meta Tutor*, *Prime Climb*, *Easy with eve*, *EER-Tutor*, *Fermat*, y *Gurú Tutor*; los autores investigan el tipo de sensores que se pueden utilizar, las emociones detectadas y las emociones factibles a reconocer por el tutor.

A nivel experimental, algunos tutores que monitorean el rostro son *Meta Tutor*, *Easy with eve* y *Auto Tutor*. *Meta Tutor* utiliza sensores de retina para la extracción de datos característicos de la mirada, lo cual le permite reconocer emociones para representarlas mediante los estados: aburrido, curioso e interesado. *Easy with eve* utiliza la videocámara para la detección y extracción de características faciales, el tutor utiliza máquinas de representación de vectores, permitiendo la identificación de estados emocionales tales como: sonriendo, riendo, sorprendido, enojado, asustado, triste, disgustado y neutral. Finalmente, *Auto Tutor* tiene la capacidad de detectar y procesar emociones a través de una videocámara utilizada para registrar el patrón de movimiento de los ojos en un estudiante. Adicionalmente, este tutor inteligente utiliza una silla sensible a la presión para la extracción de la postura del usuario. El tutor realiza sus inferencias utilizando una combinación de modelos computacionales, tales como redes neuronales, regresiones y árboles de decisión. Las emociones que se pueden representar son: confundido, aburrido, frustrado y neutral.

Por otro lado, en el ámbito del monitoreo de emociones, se encuentra *Prime Climb*, que es un tutor que utiliza sensores psicológicos por medio de la conductividad de la piel, estos sensores le permiten medir frecuencia cardíaca y la actividad muscular. Los datos biométricos se analizan para reconocer emociones como: felicidad, tristeza, admiración, crítica, orgullo y vergüenza.

Marco Teórico

El fundamento teórico que sustenta el desarrollo de esta investigación se basa en tres áreas de estudio principales:

- 1) Sistemas de Tutoría Inteligente.
- 2) Representación de conocimiento mediante Ontologías.
- 3) Agentes Inteligentes y Sistemas Multiagentes.

A continuación, se describen los fundamentos a considerar en este trabajo, derivados de cada una de las áreas de conocimiento de interés, propias de las ciencias computacionales y en específico de la inteligencia artificial educativa.

2.3. Sistemas de tutoría inteligente

Los Sistemas de Tutoría Inteligente son asesores virtuales basados en software, que surgen como una opción para incrementar el desempeño académico de los estudiantes al brindar asesoría personalizada; mediante conocimiento previamente obtenido de asesores humanos. Este tipo de sistemas han sido implementados con éxito atendiendo una diversidad de dominios (ej. matemáticas, idiomas y medicina). El desarrollo de los ITS requiere generar los tres modelos que los constituyen (dominio, estudiante y pedagógico) y los mecanismos de interacción para poder brindar asesoría personalizada a sus usuarios. A continuación, se describe brevemente el objetivo y funcionalidad de cada uno de estos modelos, así como la interacción que existe entre ellos (ver Figura 2.1).



Figura 2.1. Proceso del ciclo de asesoría del ITS: (1) captura la actividad del estudiante; (2) procesa los datos de entrada; (3) actualiza el modelo cognitivo del estudiante; (4) evalúa el desempeño del estudiante; (5) define la acción de asesoría a ofrecer; (6) selecciona el esquema de comunicación; y (7), proporciona retroalimentación al estudiante.

El Modelo del Dominio representa el conocimiento de expertos. Generalmente, mediante estructuras de representación de conocimiento, se modela el cómo solucionar problemas o ejercicios de dominios específicos, el conjunto de respuestas correctas para cada problema, errores más comunes cometidos por el estudiante y la asesoría brindada cuando estos ocurren.

EL Modelo del Estudiante, representa el nivel de conocimiento de cada uno de los usuarios del sistema. Incluye errores comunes del estudiante, tiempo requerido e intentos para solucionar un problema específico, respuestas correctas e incorrectas, así como información alterna como sus preferencias y comportamientos en el aprendizaje. Este modelo es el responsable de establecer el perfil cognitivo y aspectos que influyen en el aprendizaje de cada alumno, mediante la identificación de sus competencias y deficiencias en un área de conocimiento objetivo, formando una imagen instantánea y adaptativa de su comprensión de los contenidos.

El Modelo del Estudiante constituye una característica muy importante para los ITS, que es su capacidad de adaptación a las necesidades de aprendizaje de cada estudiante. Es decir, el sistema debe determinar el “estado cognitivo” de cada estudiante con base en sus experiencias de aprendizaje al realizar ejercicios proporcionados por el tutor inteligente. De este modo el ITS podrá determinar la estrategia de estudio más conveniente y el tipo de acción a seguir al proporcionar asesoría al estudiante, por

ejemplo, el grado de complejidad de los ejercicios y el tiempo adecuado para proporcionar asesoría.

Específicamente, se recomienda que el Modelo del Estudiante esté constituido por los siguientes submódulos que abarcan aspectos emocionales, conductuales y cognitivos [18]:

- *Estado de conocimiento*, considerando un dominio específico, es importante registrar las competencias iniciales (con base en el Modelo del Dominio), su estado cognitivo actual y preferentemente un historial indicando cómo ha evolucionado entre los diferentes niveles de conocimiento logrados.
- *Estilo de aprendizaje*, considerado como la forma de clasificar el comportamiento de un estudiante de acuerdo a la manera en que toma la información y forma las estrategias para aprender, cómo entiende y cómo le gusta analizar la información que está utilizando para acceder a un conocimiento determinado. En otras palabras, es una forma de agrupar o clasificar a un estudiante de acuerdo a su perfil de aprendizaje, considerando que este estilo evoluciona y cambia de acuerdo a las variables del entorno que afectan en su aprendizaje.
- *Perfil psico-sociológico del estudiante*, aquí se determinan las debilidades y las fortalezas que tiene un individuo en su aprendizaje, considerando aspectos conductuales, de interacción, hábitos de estudio, entre otros.
- *Historial de errores*, guarda en la base de datos los errores que comete el estudiante al tratar de solucionar determinado ejercicio.
- *Historial de consultas*, almacena cada consulta o solicitud de ayuda que el estudiante realiza al tratar de solucionar determinado ejercicio.
- *Historial de decisiones*, registra en la base de datos cada decisión o acción que el estudiante toma (ej. participación en foros, visualización de videos, revisión de material introductorio, etc.).

En el Modelo Pedagógico se encuentran almacenadas las estrategias de enseñanza y asesoría aprendidas por el asesor inteligente. Considerando entre estas estrategias, el tipo de asesoría a proporcionar, mecanismos para determinar cuándo y cómo brindar

asesoría, así como la cantidad apropiada de retroalimentación que debe recibir cada estudiante.

El módulo de Comunicación, es el que permite el uso de la interfaz del ITS, aquí el asesor inteligente monitorea y registra el desempeño del estudiante mientras éste resuelve ejercicios e interactúa con el sistema, ofreciéndole la asesoría adecuada en el momento oportuno. La información de esta interacción se va registrando en el ITS, y este va determinando cómo enseñar a cada estudiante ya que cada uno aprende de manera distinta y posee un nivel diferente de conocimiento.

En la presente investigación, se estudia y desarrolla un modelo del estudiante que representa aspectos emocionales y conductuales, el cual, al cubrir aspectos genéricos de un alumno, puede ser compartido entre tutores inteligentes de diversos dominios.

2.4. Representación de conocimiento mediante Ontologías

El término ontología es una rama filosófica dirigida al estudio del ser. Siendo un término que deriva del griego “ontos”, que significa “ser”, y “logos”, que tiene el significado “estudio de” [32]. En el ámbito de las ciencias computacionales, una ontología es la representación del conocimiento de un dominio específico mediante clases (entes) y las relaciones entre ellas [33]. De igual forma, una ontología, en [34], es considerada como una especificación de una conceptualización, es decir, es una estructura sistematizada que no solo almacena información, adicionalmente permite la búsqueda y recuperación de la misma. Siendo esta forma de representación de la realidad una opción con potencial para representar la base de conocimiento de ambientes inteligentes, particularmente para el modelado del conocimiento de expertos y/o del estudiante; propios de los sistemas de tutoría inteligente.

En [35], se consideran los siguientes elementos para el desarrollo de una ontología:

Conceptos: Describe las clases a representar: estrategias, planes u objetos.

Relaciones: Indica las interacciones entre las clases del dominio específico.

Funciones: Es un tipo de relación utilizada para indicar acciones entre elementos (ej. categorizar-clase, asignar-fecha).

Instancias: Se utilizan para la representación de objetos específicos de una clase.

Axiomas: Son teoremas que se declaran en las relaciones, indicando requisitos que deben cumplir las clases de la ontología.

2.4.1. Lenguajes ontológicos

La forma de implementar ontologías es mediante el uso de lenguajes ontológicos. Generalmente, a través de este tipo de lenguajes, se especifica la ontología que es la base de conocimiento de algún sistema. Mediante estos lenguajes se almacenan declaraciones sobre las relaciones entre entidades utilizando alguna especificación como el *Marco de Descripción de Recursos* (RDF, por sus siglas en inglés) y se utiliza un razonador que nos permita inferir entre los datos censados y la base de conocimiento.

2.4.2. RDF

En [36], se define al RDF como un modelo de representación de metadatos que puede utilizarse como lenguaje general para el modelado del conocimiento. Al emplear las estructuras RDF se utilizan triples para especificar relaciones o expresiones del tipo *sujeto-predicado-objeto*. Toda triple es considerada como una sentencia RDF, el *recurso predicado* es un identificador, que deriva del *recurso sujeto*, que determina un tipo de propiedad o relación que se desea establecer acerca del actor o sujeto. El *recurso objeto* identifica el valor del *recurso predicado*. Estos recursos requieren del uso de *Identificadores de Recursos Uniformes* (URI, por sus siglas en inglés), que son identificadores personalizados y únicos en Internet, definidos por la combinación de un dominio de Internet y el nombre asignado por la persona que los crea.

Los siguientes son ejemplos de un identificador URI y una expresión RDF, en ese orden. El primero de ellos representa dirección, referida como el identificador único, de uno de los recursos de vídeo disponible en la plataforma educativa RITADA; dirigido a fortalecer estrategias de concentración al estudiar. El segundo ejemplo representa la asignación del atributo *título* a este vídeo, utilizando la propiedad *title* del vocabulario *Dublin Core Metadata Element*, definido por el modelo de metadatos propuesto por la

organización *Dublin Core*, creado para operar mediante una descripción homogénea y garantizar la localización de recursos digitales en la Web [36]. Más adelante se incluyen otros ejemplos que contienen tripletas de RDF.

URI: <https://ritada.ens.uabc.mx/servicio/siec/mod/hvp/view.php?id=160>

RDF: <https://ritada.ens.uabc.mx/servicio/siec/mod/hvp/view.php?id=160>

<<http://purl.org/dc/elements/1.1/title>>

“Concentración”

2.4.3. Razonador

Para llevar a cabo consultas en una ontología es necesario contar con un razonador de lógica descriptiva (DL, por sus siglas en inglés). Este tipo de razonadores proveen servicios para extraer conclusiones de conocimiento representado en ontologías [37]. Aunque existen diversos razonadores en la literatura, el razonador empleado en esta investigación fue FACT++, el cual es uno de los más utilizados por la comunidad en esta área de estudio ya que permite trabajar con herramientas que soportan el estándar de lenguajes ontológicos OWL.

2.4.4. Modelos ontológicos orientados al estudiante

En la literatura existen modelos ontológicos orientados a la educación, como el propuesto en [38], donde se describen las características que se consideran importantes para representar aspectos conductuales y cognitivos del alumno. En este modelo, el autor da énfasis al modelado del estilo de aprendizaje de los estudiantes, considerando 4 categorías binarias (*sensorial-intuitivo*, *visual-verbal*, *activo-reflexivo* y *secuencial-global*), que pueden generar 16 combinaciones posibles (ver Figura 2.2).

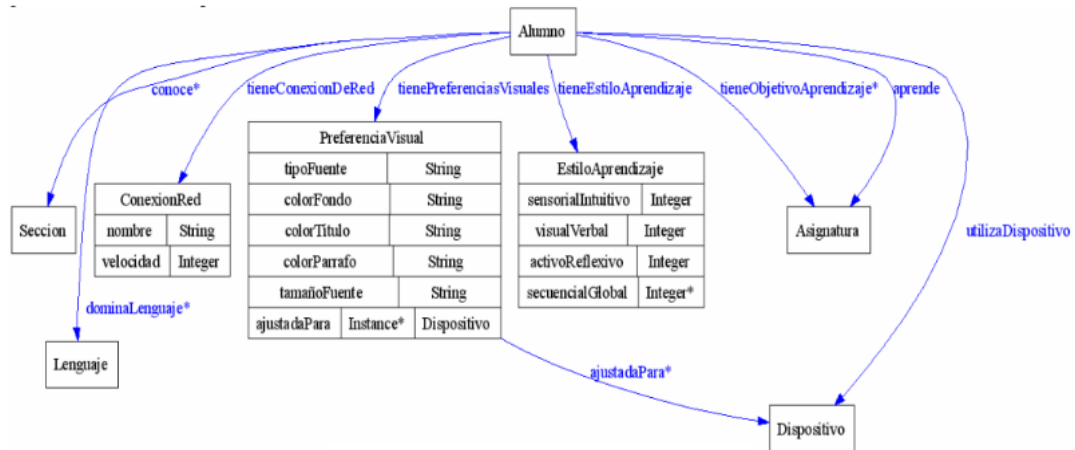


Figura 2.2. Ontología representando el modelo del estudiante [38].

Otra aportación es la realizada en [39], del Instituto de Matemáticas e Informática B.A.S., Bulgaria, donde se utilizó una ontología para representar el modelo del estudiante de un ITS desarrollado para apoyar en el uso de bibliotecas digitales. Otra ontología revisada fue la ontología propuesta en [40], donde se aborda la investigación de un modelo ontológico útil para el desarrollo de sistemas de tutoría inteligente a distancia. Los autores proponen el uso del perfil del estudiante como un módulo de un tutor inteligente, en el que se hace la integración de reglas semánticas combinadas con mecanismo de inferencia para la clasificación de los estudiantes de acuerdo a sus cursos, actividades y su información personal, Como resultado de esta investigación se propone el módulo para ser integrado en aplicaciones Web para la enseñanza. El estudio fue desarrollado en la universidad de Patras Grecia. En estas dos investigaciones se le da énfasis al modelado de los aspectos cognitivos, prevaleciendo ante los aspectos conductuales.

En resumen, una ontología es un conjunto de símbolos o términos, aunados a su correspondiente interpretación o significado, y la relación entre ellos. Cuando dos agentes o tutores inteligentes usan una misma ontología se asegura de que sus interpretaciones de lo que se está representando coinciden. Con este antecedente se determinó elaborar una ontología para organizar y proporcionar la información a manipular por el ambiente integral de tutores propuesto.

Los beneficios de utilizar esta ontología se pueden resumir de la siguiente forma:

- proporcionan una forma de representar y compartir el conocimiento utilizando un vocabulario común,
- permiten usar un formato estandarizado para el intercambio de conocimiento,
- proporcionan un protocolo específico de comunicación,
- permiten una reutilización del conocimiento.

2.5 Comunicación multiagente

En el ámbito de la Inteligencia Artificial, un Agente Inteligente (IA, por sus siglas en inglés) es “*cualquier ente capaz de percibir su ambiente con la ayuda de sensores y actuar en ese medio utilizando actuadores*” [41]. Pudiendo resumir la esencia de un IA con su capacidad de interactuar en un entorno dado y realizar tareas específicas, siempre con el fin de lograr un objetivo de la mejor manera posible. En particular, los tutores inteligentes, considerados como un tipo de agentes pedagógicos, surgen como una opción para la enseñanza cuyo objetivo principal es incrementar el desempeño académico de los estudiantes al brindarles asesoría personalizada; utilizando una base de conocimiento previamente obtenida de asesores humanos.

Por otro lado, los *sistemas multiagentes* [42] refieren a universos artificiales, donde un conjunto de IA interactúa y/o colabora para lograr un objetivo común. Una metodología particular para construir sistemas multiagentes especifica cómo el agente puede ser descompuesto en un conjunto de módulos, y cómo estos módulos pueden interactuar entre sí. El conjunto total de módulos y sus interacciones deben proveer una respuesta a la pregunta de cómo los datos monitoreados, y el estado interno de cada uno de los agentes (módulos) determinan las acciones a realizar.

Un ejemplo de arquitectura multiagente para la implementación de tutores inteligentes es la utilizada en el desarrollo del *ITS ADVISOR* [43]. Sistema compuesto por dos agentes cuyo objetivo es centralizar el razonamiento de un ITS al decidir el tipo de asesoría. Uno de los agentes es responsable de actualizar el modelo del estudiante al monitorear el comportamiento de los alumnos que usan el tutor. El otro agente toma este modelo de comportamiento del alumno y analiza el objetivo que especifica la

competencia educacional que se persigue, con base en ello se determina el tipo de retroalimentación o seguimiento de la instrucción del estudiante. Sin embargo, el desarrollo de esta arquitectura no considera la interacción entre ITS que cubren diversos dominios, solo ve a los agentes como módulos internos e independientes dentro del tutor específico.

Entre las arquitecturas para la construcción de sistemas multiagentes se encuentran las *deliberativas*, *reactivas*, e *híbridas*, a continuación, se describe cada una de estas:

Las arquitecturas deliberativas son aquellas arquitecturas que utilizan modelos de representación simbólica del conocimiento [44]. El modelado de conocimiento humano se realiza mediante una descripción textual que posteriormente es consumido utilizando reglas de inferencia que permiten al agente tomar las decisiones [45]. Este tipo de arquitecturas suelen estar basadas en la teoría clásica de planificación, es decir, estos agentes parten de un estado inicial y son capaces de generar planes para alcanzar sus objetivos.

Las arquitecturas reactivas se caracterizan por no tener como elemento central de razonamiento un modelo simbólico; generalmente se determinan las tareas a realizar de forma algorítmica [46]. En estas arquitecturas se tiene una respuesta inmediata a cambios del entorno; infiriendo o estimando la mejor acción posible.

Mediante las arquitecturas híbridas se construye un agente compuesto de dos subsistemas: uno deliberativo, que utilice un modelo simbólico del conocimiento y que genere planes, y otro reactivo, centrado en reaccionar a cambios en el entorno y que no necesite mecanismos complejos de razonamiento [47].

Las principales arquitecturas que se describen nos permiten observar el comportamiento de un agente, desde perspectivas tales como la deliberativa y la reactiva, donde ambas arquitecturas nos muestran al agente en su idea básica. Particularmente, en la arquitectura deliberativa se muestra la necesidad que tiene un agente de razonar, con base en conocimiento previamente establecido, para poder reaccionar adecuadamente a la situación en la que se encuentre. Sin embargo, revisando las arquitecturas utilizadas para el desarrollo de tutores inteligentes, no se encontraron arquitecturas que soporten el compartir módulos entre agentes interactuando en diferentes dominios o contextos, se

observó que los agentes empleados en cada arquitectura son agentes “autónomos” que no comparten sus recursos, solo comparten el entorno de operación (el Internet), infieren el comportamiento del resto de los agentes y con base en esto toman decisiones.

Para la implementación del ambiente propuesto se utilizó un modelado de conocimiento basado en ontologías, y una máquina de inferencia basada en reglas. Con base en lo anterior, la arquitectura multiagente propuesta está considerada dentro del esquema deliberativo. En la Figura 2.3 se muestra el ambiente planteado que permite la comunicación entre tutores, sin importar el dominio al que pertenezcan, e implementados por diversas fuentes. En este caso particular, provenientes de diferentes instituciones de educación superior; donde las instituciones se representan con las iniciales “IES”.

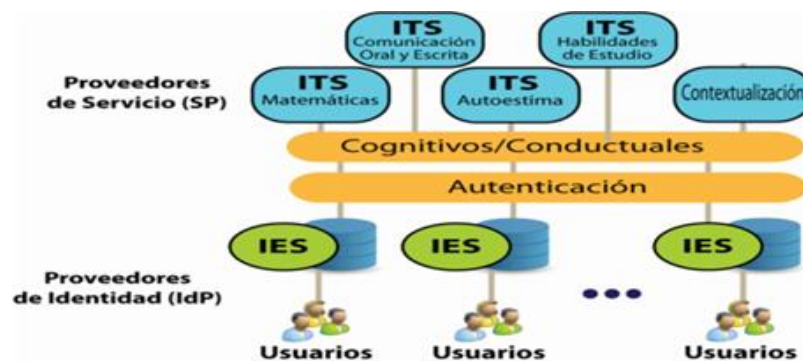


Figura 2.3. Arquitectura de servicios de tutoría inteligente en un ambiente integrado.

La coordinación es otro elemento principal en un sistema multiagentes, siendo una actividad que implica definir el esquema para el manejo de interdependencias al momento de que los agentes realizan sus respectivas tareas [48]. Una tipología para clasificar agentes refiere a una relación *positiva* o *negativa*. La primera refiere al hecho de que los agentes que interactúan en un ambiente obtienen beneficios, ya sea uno de ellos o ambos, pero no se perjudican entre ellos. Mientras que en un sistema multiagente del tipo negativo, uno de los agentes puede verse afectado en el logro del objetivo del contrario; situación que se presenta en ambientes de competencia multiagente. En nuestro caso particular, el tipo de relación entre agentes es positiva.

Al mismo tiempo, dentro de un esquema de relación positiva, la interacción entre agentes puede ser explícita o implícita. Es decir, un agente puede solicitar explícitamente la ayuda de otro agente para poder llevar a cabo su siguiente tarea o, se puede ver beneficiado de los productos de otro agente sin necesidad de solicitarlos. El tipo de colaboración de nuestro interés está fuertemente identificado como una interacción implícita y por consecuencia, donde el agente cuyo objetivo principal es generar el modelo del estudiante lo más cercano a la realidad, beneficia directamente al objetivo del agente responsable del modelo pedagógico, cuya meta es proporcionar instrucción y retroalimentación personalizada.

Por otra parte, el sistema integral de tutoría inteligente propuesto puede ser considerado como un *ecosistema digital de aprendizaje*. Este tipo de sistemas se componen de un conjunto de actores, dispositivos y herramientas de software que interactúan entre sí y producen flujos de información que propician el aprendizaje [49]. Aun cuando el término ecosistema digital fue inicialmente acuñado en el ámbito empresarial y se orientaba a la generación de redes de colaboración entre organizaciones, este concepto fue posteriormente adoptado en el entorno educativo bajo el nombre de ecosistema de aprendizaje [50].

En la revisión de la literatura se encontraron algunos estudios que hacen referencia a las características de un ecosistema de aprendizaje, así como los mecanismos que permiten la interacción entre sus componentes. Una de las primeras publicaciones sobre ecosistemas de aprendizaje es la investigación de [51], en la cual se analizan las limitaciones de los sistemas convencionales dirigidos a la enseñanza. En este estudio se describe cómo un ecosistema digital de aprendizaje puede ayudar a integrar sistemas y dispositivos heterogéneos, lo que permite su interacción y favorece el aprendizaje.

Recientemente, en el trabajo publicado en [52] se plantea la necesidad de un entorno tecnológico para ofrecer servicios educativos que permitan obtener mejores resultados en los procesos de enseñanza-aprendizaje. Asimismo, se propone un ecosistema educativo que se conforma de seis elementos:

- 1) una arquitectura para la inserción del ecosistema de aprendizaje,
- 2) realizar toma de decisiones basada en analíticas de aprendizaje,
- 3) implementar sistemas de gestión de conocimiento adaptativos,

- 4) incluir formación lúdica,
- 5) utilizar portafolios semánticos para guardar evidencias de aprendizaje, y
- 6) basarse en metodologías educativas para la mejora del aprendizaje.

Similar a lo que plantearon en [50], los autores concluyen que el simple uso de un sistema de gestión de aprendizaje (LMS, por sus siglas en inglés) ya no es suficiente para sustentar el aprendizaje en línea, sino que se deben implementar ecosistemas de aprendizaje que integren y comuniquen todas las herramientas tecnológicas orientadas a la enseñanza y permitan poner en funcionamiento estrategias pedagógicas personalizadas, las cuales atiendan las necesidades específicas de una comunidad educativa y se puedan adaptar a cualquier dominio. Por otro lado, en la búsqueda de generalizar el diseño y modelado de ecosistemas de aprendizaje, la investigación en [51] describe los requerimientos de alto nivel de un ecosistema de aprendizaje y propone un metamodelo para el diseño de este tipo de ambientes de aprendizaje.

El metamodelo propuesto en [52] está basado en la *Arquitectura Dirigida por Modelos* (MDA, por sus siglas en inglés) que, a través de guías para estructurar especificaciones de procesos, busca separar el diseño de flujos funcionales de las tecnologías que los implementan [53]. El metamodelo se validó al representar un ecosistema de aprendizaje real que estaba orientado a la gestión del conocimiento científico de estudiantes de doctorado. Los autores aseveran que este metamodelo puede ser transferido a otros dominios y fácilmente convertirse en un prototipo funcional.

Un caso práctico, donde se utilizan tutores inteligentes como parte de un ecosistema de aprendizaje, se realizó en la investigación de [54]. Los investigadores implementaron un ecosistema de aprendizaje utilizando el marco inteligente generalizado para tutoría (GIFT, por sus siglas en inglés), software de libre uso que permite la autoría de tutores inteligentes [55]. GIFT incluye un LMS para la gestión de contenidos y una extensión de reciente liberación que permite interactuar con la especificación de Experience-API- [56].

En conclusión a este capítulo, el desarrollo del Sistema Integral de Tutoría Inteligente propuesto se fundamenta fuertemente en los principios funcionales estipulados para un ecosistema de aprendizaje. Adicionalmente, el ambiente para la asesoría académica desarrollado en esta investigación, integra el modelado de

conocimiento mediante lenguajes ontológicos, que facilita la interacción entre agentes pedagógicos multidominio y multiplataforma.

Capítulo 3

Metodología

En la presente investigación se atiende un método de estudio cuantitativo, ya que se analizaron y definieron técnicas para la implementación y evaluación de funcionalidad y aceptabilidad de una arquitectura núcleo para la cooperación entre tutores inteligentes. Basada en un modelo del estudiante compartido, implementando mecanismos de integración con modelos de experto independientes del dominio. Por lo anterior, considerando ciertas métricas de desempeño de software definidas por el modelo de calidad ISO/IEC 9126-1, se analizó el desempeño de la arquitectura propuesta con base en su nivel de:

- **operatividad**, capacidad de intercambio de información entre distintos sistemas.
- **Rendimiento**, tiempo de respuesta y uso adecuado de recursos.
- **Compatibilidad**, integración y capacidad de coexistencia.
- **Escalabilidad**, capacidad de manipular múltiples tutores inteligentes.

El paradigma de la investigación se define como una investigación exploratoria ya que se aborda un tópico poco estudiado, como es la arquitectura cooperativa entre ITS para la interacción entre tutores inteligentes y su nivel de aceptación por los usuarios.

La metodología propuesta en la presente investigación está constituida por tres etapas principales: a) *modelado de conocimiento y desarrollo tecnológico*: consistente en crear la estructura que permita modelar el conocimiento del estudiante y diseñar las capas de la arquitectura integral de tutores inteligentes de dominios diversos; b) *implementación del protocolo de comunicación entre tutores*: definición e integración del protocolo de comunicación que permita la interacción de tutores inteligentes de dominios diversos; y c) *evaluación del desempeño y nivel de aceptación de la arquitectura de software propuesta al interactuar con múltiples tutores inteligentes provenientes de diversos dominios*: medir el desempeño del ambiente propuesto con base en los criterios de operatividad, rendimiento, compatibilidad y escalabilidad, así como el nivel de aceptación de la asesoría brindada.

3.1 Modelado de conocimiento y desarrollo tecnológico

La fase de desarrollo de la arquitectura de software propuesta, en apoyo a la comunicación entre tutores inteligentes, considera dos secciones principales: 1) modelado del conocimiento del usuario basado en ontologías; y 2) diseño de las capas de la arquitectura integral de tutores inteligentes para la interacción entre el modelo de experto y el modelo del estudiante de tutores inteligentes.

3.1.1 Modelado de conocimiento basado en ontologías

En el desarrollo de esta etapa, para responder a nuestra primera pregunta específica de investigación *¿Qué esquema de representación de conocimiento permite la implementación del modelo del estudiante que facilite su integración con modelos de experto independientes del dominio?*, se tomó la decisión de utilizar técnicas de modelado de conocimiento mediante semántica ontológica.

Al tratarse de un problema complejo, en el que se necesita representar el conocimiento del estudiante de una forma que pueda ser validado por expertos humanos y que sea entendido y manipulado por los sistemas computacionales, se opta por el uso de modelos ontológicos. Al mismo tiempo, esta modalidad permite al tutor inteligente ejecutar algoritmos que inciden en la toma de decisiones para brindar asesoría personalizada a los alumnos que interactúan con el sistema, sin importar en qué ambiente o lenguaje de desarrollo esté programado.

Específicamente, se diseñó la ontología en el software Protege 5.0, sistema para administrar conocimiento mediante su editor de ontologías (ver Figura 3.1). Protege es un sistema de código libre, con una diversidad de funcionalidades para el modelado, generación, y uso de ontologías; lo cual lo ha llevado a ser uno de los ambientes de desarrollo más recomendados en la literatura.

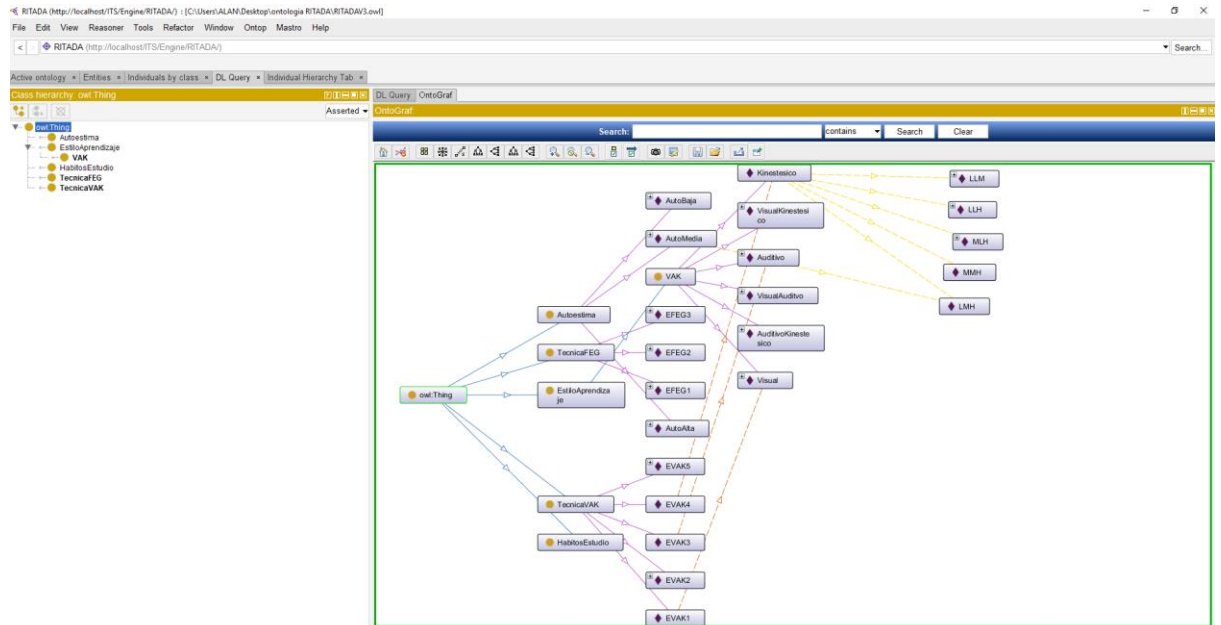


Figura 3.1. Protege: ambiente para la creación de ontologías.

La ontología implementada en el ambiente propuesto contiene seis clases principales: *Estilo de aprendizaje*, basada en el modelo VAK (Auditivo, Kinestésico y Visual); *Hábitos de estudio*, *Autoestima* (Familiar, Escolar y General - FEG), *Técnica FEG*, *Técnica VAK* y *Actividad* (ver Figura 3.2).

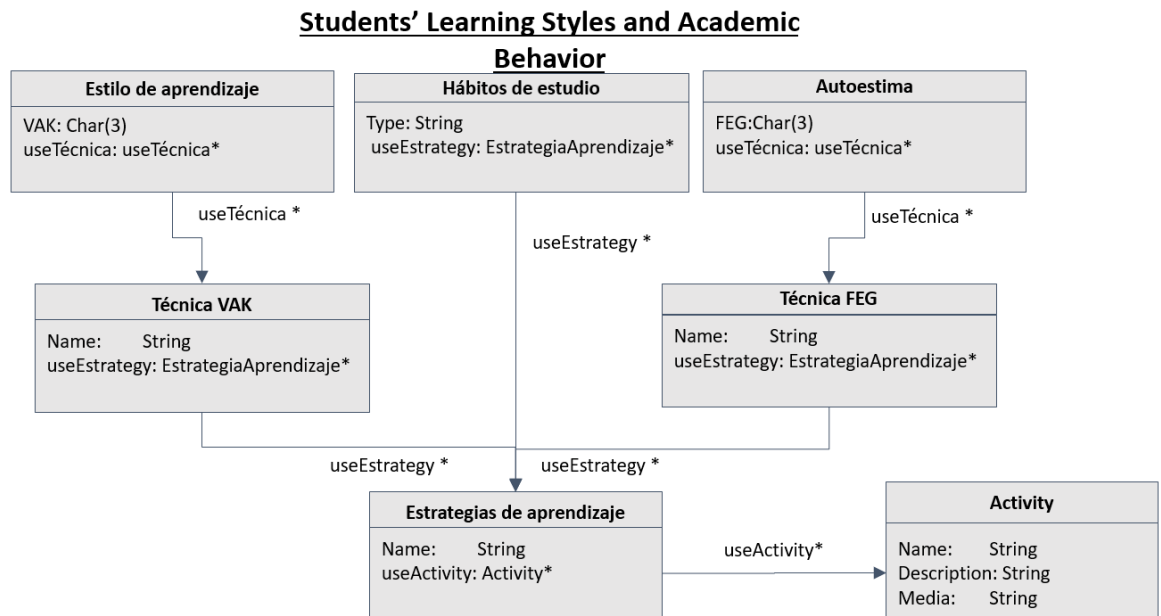


Figura 3.2. Nivel principal de la ontología representando el modelo del estudiante pedagógico.

En la clase *Estilo de aprendizaje* se generan las combinaciones de estilos de aprendizaje *kinestésico*, *visual* y *auditivo*, las cuales pueden ser categorizadas mediante la escala alta, media y baja: H, M y L, respectivamente. Para la clase *Study_Habits* se toman en cuenta los aspectos de *concentración*, *relaciones personales*, *memoria*, *motivación*, *administración de tiempo*, y *pruebas*. En la clase *Autoestima* se registra el nivel de *autoestima general*, *autoestima escolar* y *autoestima familiar*. Similar a la clase *estilo de aprendizaje*, se generan las combinaciones con escala H, M y L. La clase *Técnica FEG* y *Técnica VAK* contienen las estrategias de aprendizaje que deberá utilizar el tutor inteligente y la clase *Actividad* contiene las actividades a asignar al alumno. Las clases que representan los estilos de aprendizaje y autoestima, reciben la combinación proveniente de la plataforma RITADA.

Una vez establecidas y creadas las clases se definieron las relaciones de la ontología. Es fundamental establecer las relaciones ya que esto es lo que permite al razonador tomar la mejor decisión. En la Figura 3.2 se aprecia la relación jerárquica implementada en el modelo de representación de conocimiento propuesto. Posteriormente, se procedió a la generación de las instancias en cada clase. Para la clase *autoestima* se declararon las instancias *AutoAlta* (Autoestima Alta), *AutoMedia* (Autoestima Media) y *AutoBaja* (Autoestima Baja). Para la instancia *AutoAlta* se definieron sus combinaciones con las escalas HHL, HHM, LHH, MHH, HHH, LHL, LHM, MHL y MHM (se consideran combinaciones donde el segundo componente FEG es alto– Escolar). En el caso de la instancia *AutoMedia* se definieron sus combinaciones con las escalas HML, LMH, HMM y LML (se consideran combinaciones donde el segundo componente FEG es medio– Escolar). Al igual que en las instancias anteriores, en *AutoBaja* se declararon sus combinaciones del tipo HLL, HLM, MLM, LLL, MLL, LLM, HLH, LLH y MLH (se consideran combinaciones donde el segundo componente FEG es bajo– Escolar). Las declaraciones de las combinaciones para la ontología propuesta se llevaron a cabo con la palabra reservada *Is*, generando la palabra *IsCombinación*, que es la finalmente utilizada para crear estas relaciones en el desarrollo de la ontología.

Adicionalmente, se declaran las instancias para la clase *Técnica FEG* donde están contenidos las tres distintas técnicas de enseñanza propuestas que son: EFEG1, EFEG2 y EFEG3. Estas técnicas o estrategias de enseñanza son las que se le asignan al tutor

inteligente para que brinde la asesoría de forma personalizada. Para la instancia EFEG1 se define como propiedad de afirmación (*object-property-assertion*) a la palabra reservada *HasNivelAutoestima* del tipo *AutoAlta*. Lo anterior implica que cualquier combinación que esté en la clase de autoestima alta recibirá la estrategia EFEG1. Al mismo tiempo se declara como valor de esta afirmación (*data-property-assertion*) la palabra VALOR, asignándole el número 1, como tipo entero, para que nuestra ontología devuelva este valor a los algoritmos implementados en el tutor inteligente.

Para la instancia EFEG2 se define como *object-property-assertion* la palabra reservada *HasNivelAutoestima* del tipo *AutoMedia*, esto quiere decir que cualquier combinación que esté en la clase autoestima media recibirá la estrategia EFEG2. Al mismo tiempo se declara como *data-property-assertion* la palabra VALOR con el número 2 para que nuestra ontología devuelva este valor al tutor inteligente. Finalmente, para la instancia EFEG3 se define como *object-property-assertion* la palabra reservada *HasNivelAutoestima* del tipo *AutoBaja*, esto quiere decir que cualquier combinación que esté en la clase autoestima media recibirá la estrategia EFEG3, declarando como *data-property-assertion* la palabra VALOR con el número 3.

Por otra parte, cabe mencionar que se declararon las instancias en las distintas clases definidas en la ontología, para la clase estilo de aprendizaje se describe la subclase VAK en la que están contenidos los estilos de aprendizaje definidos como instancias: *Visual*, *Auditivo*, *Kinestésico*, *VisualAuditivo*, *VisualKinestésico* y *AuditivoKinestésico*. Posteriormente, se declararon las combinaciones para cada instancia. Para la instancia *Visual* se consideraron las combinaciones con las escalas: MLL, HLM, HML, HMM y HLL (se consideran combinaciones donde el primer componente VAK es alto o medio con los otros componentes bajos– Visual). Para la instancia *Auditivo* se declararon sus combinaciones con las escalas: MHM, LHM, LHL, MHL y LML (se consideran combinaciones donde el segundo componente VAK es alto o medio con los otros componentes bajos– Auditivo).

Para la instancia *Kinestésico* se declararon sus combinaciones con las escalas: LMH, MLH, LLH, MMH y LLM (se consideran combinaciones donde el tercer componente VAK es alto o medio con los otros componentes bajos– Auditivo). Para la instancia *VisualAuditivo* se declararon sus combinaciones con las escalas: MML, HHL y HHM (se consideran combinaciones donde el primer y segundo componente VAK son

idénticos, alto o medio, con el tercer componente bajo– VisualAuditivo). Para la instancia *VisualKinestésico* se declararon sus combinaciones con las escalas: HLH, MLM y HMH (se consideran combinaciones donde el primer y tercer componente VAK son idénticos, alto o medio, con el segundo componente bajo– VisualKinestésico). Para la instancia *AuditivoKinestésico* se declararon sus combinaciones con las escalas: MHH, LHH y LMM (se consideran combinaciones donde el segundo y tercer componente VAK son idénticos, alto o medio, con el primer componente bajo– AuditivoKinestésico).

También, se procedió a definir las instancias de la clase *Técnica VAK*: EVAK1, EVAK2, EVAK3, EVAK4 y EVAK5. Para la instancia EVAK1 se indicó el object-property-assertion con la palabra reservada *HasEstilo* del tipo *Visual*, esto indica que cualquier combinación que esté en la clase de estilo de aprendizaje visual recibirá la estrategia EVAK1, al mismo tiempo devuelve un 1 como valor entero al tutor inteligente. Para la instancia EVAK2 se define como object-property-assertion la palabra reservada *HasEstilo* del tipo *VisualAuditivo*, esto quiere decir que cualquier combinación que esté en la clase de estilo de aprendizaje visual auditivo recibirá la estrategia EVAK2, devolviendo un 2 como valor entero al tutor inteligente.

En la instancia EVAK3 se define como object-property-assertion la palabra reservada *HasEstilo* del tipo *Auditivo*, esto quiere decir que cualquier combinación que esté en la clase Estilo de aprendizaje auditivo recibirá la estrategia EVAK3, la cual devuelve un 3 como valor entero. Para la instancia EVAK4 se define como object-property-assertion la palabra reservada *HasEstilo* del tipo *Kinestésico*, esto quiere decir que cualquier combinación que esté en la clase estilo de aprendizaje kinestésico recibirá la estrategia EVAK4, devolviendo un 4 como valor entero. La instancia EVAK5 se define como object property assertion la palabra reservada *HasEstilo* del tipo *AuditivoKinestésico*, esto quiere decir que cualquier combinación que esté en la clase estilo de aprendizaje auditivo kinestésico recibirá la estrategia EVAK5, que devuelve un 5 como valor entero. Una vez procesados los datos provenientes del diagnóstico de la plataforma RITADA el razonador de nuestra ontología envía estos datos numéricos para que el tutor inteligente asigne la asesoría personalizada a través de la plataforma RITADA.

3.1.2 Diseño de capas de la arquitectura integral de tutores inteligentes

En esta etapa del estudio, atendiendo nuestra segunda pregunta de investigación *¿Qué modelo de arquitectura de software permitiría la cooperación e interacción entre múltiples tutores inteligentes, al compartir su modelo del estudiante?*, se determinó el desarrollar una arquitectura integral que permita la comunicación entre tutores inteligentes heterogéneos. La decisión proviene de la revisión de la literatura en la que no encontramos una arquitectura como tal que permita la comunicación de tutores inteligentes que atienden dominios distintos. Es aquí donde se requiere del diseño de las capas que conformarán nuestra arquitectura, la que estará compuesta de actores, clientes y servidores. En esta arquitectura, los actores son los usuarios (ej. alumnos, docentes, padres o tutores), los clientes son las interfaces con las que interactúan los usuarios con los servicios (asesores inteligentes multidominio), y los servidores son los proveedores de servicios educativos a través de la integración de aplicaciones (módulos del sistema, librerías, recursos de aprendizaje y un gestor de aprendizaje digital propietario).

Particularmente, el diseño de la arquitectura integral de tutores inteligentes se fundamentó en un metamodelo basado en la *Arquitectura Dirigida por Modelos*, separando los flujos funcionales (servicios) de las tecnologías que los implementan [57, 58]. Adicionalmente, la implementación de estos servicios se sustentó en una arquitectura de comunicación multiagentes que atienden las siguientes características operativas: toma de decisiones deliberativa, relación positiva e interacción implícita; tal como se muestra en la Figura 3.3.

La característica deliberativa de la arquitectura propuesta se realizó mediante el modelado de conocimiento basado en ontologías y una máquina de inferencia basada en reglas. La coordinación en la comunicación entre los agentes del sistema, considerando la naturaleza del ambiente implementado, implica una obtención de beneficios mutuos, ya sea de uno de ellos o de más de uno, pero sin perjudicarse entre ellos. Adicionalmente, considerando que los recursos o información que genere cada agente se pondrán a disposición de los demás, para que estos sean consumidos cuando sean requeridos, la interacción se plantea ser de carácter implícito. Lo anterior nos lleva a la materialización de un ecosistema de aprendizaje orientado a la gestión de contenidos educativos y bases de conocimientos (experto, estudiante y pedagógico) en atención del fortalecimiento de competencias del estudiante en diversos dominios.

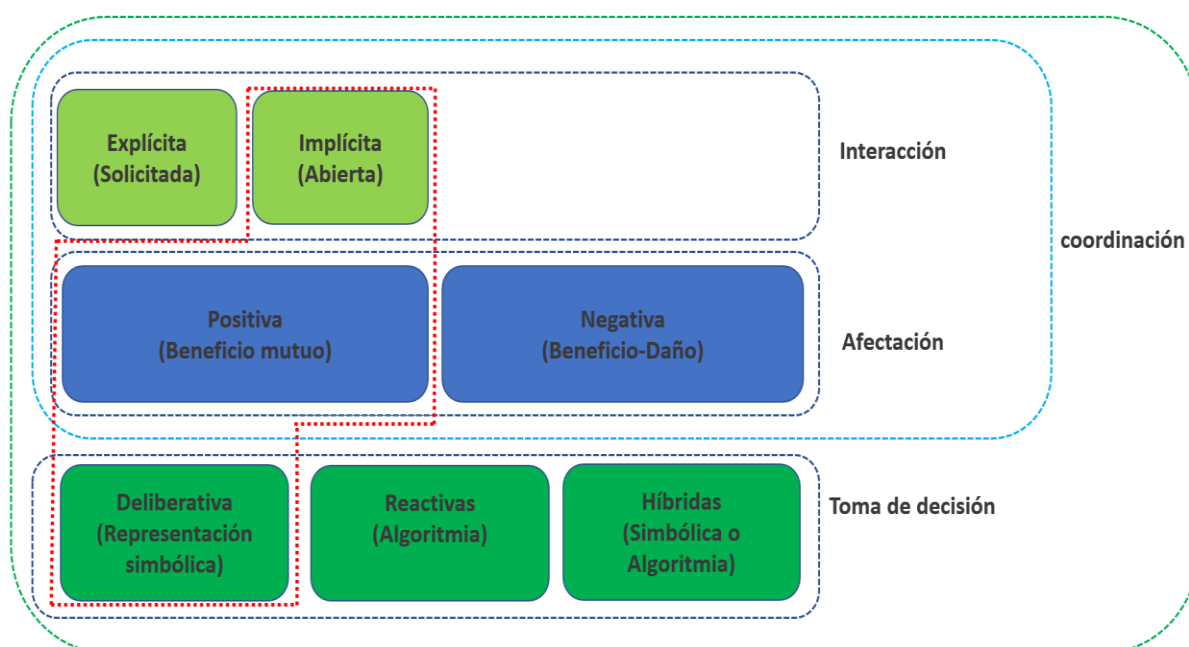


Figura 3.3. Arquitectura multiagentes propuesta (Elaboración propia)

3.2 Implementación del protocolo de comunicación entre tutores

A través del tiempo, el uso y desarrollo de las TIC en la educación han evolucionado en forma paralela. Día a día surgen nuevas tecnologías aplicadas a la educación y nuevas tendencias en su uso, impactando positivamente en el proceso de enseñanza-aprendizaje. La sociedad actual genera cada vez más información al utilizar los ambientes virtuales de aprendizaje disponibles (LMS, proveedores de cursos MOOC, Blogs educativos, entre otros), mientras que el avance tecnológico permite hospedarlos, operarlos, interconectarlos y manipular los datos generados de una forma eficiente; facilitando la labor docente, del alumno y de los directivos.

La evolución de las TIC ha provocado un crecimiento en los bancos de datos en el sector educativo. La diversidad de ambientes virtuales de aprendizaje, generalmente basados en tecnología Web, ofrecen servicios educativos que hacen posible la interacción educativa entre los diversos actores participantes: alumno-alumno, profesor-alumno, profesor-padres, profesor-directivos, etc. Algunos de los términos utilizados para categorizar a los distintos ambientes educativos son los siguientes:

(VLE) Entornos virtuales de aprendizaje.

(LMS) Sistemas de gestión de aprendizaje.

(CMS) Sistema de gestión de cursos.

(MLE) Ambiente controlado de aprendizaje.

(ILS) Sistema integrado de aprendizaje.

(LSS) Sistema soporte aprendizaje.

(LP) Plataforma de aprendizaje.

(ITS) Sistema de tutoría inteligente.

La historia de los ambientes virtuales de aprendizaje se detona en Europa en los años sesenta, con el desarrollo de unidades didácticas donde el docente se presentaba con el rol de tutor o guía y los estudiantes interactuaban para compartir y obtener conocimientos. Pero lo que realmente propició esta forma de intercambio cognitivo fue el surgimiento de Internet, al diversificar los formatos de contenidos educativos (multimedia), eliminando restricciones de tiempo y ubicación física, propiciar la reutilización de contenidos, etc. Lo anterior provocó que instituciones educativas de los distintos niveles académicos adoptaran diversos ambientes de aprendizaje, destacando los LMS como principales plataformas para la impartición de cursos a distancia [59].

La disponibilidad de distintos ambientes virtuales de aprendizaje ha permitido que las instituciones diversifiquen los servicios educativos ofrecidos; pero al mismo tiempo se genera heterogeneidad en las arquitecturas, lenguajes y estructuras de datos. Esta diversidad tecnológica deriva en complicaciones al tratar de integrar las funcionalidades proveídas, así como la compartición de los datos generados por las mismas. Esto último es considerado un factor importante, ya que, al contar con información de un mismo estudiante, generada por aplicaciones que atienden diversas áreas de estudio, permite ofrecer servicios educativos personalizados de acuerdo a las competencias generales de los usuarios.

Existen diversos esfuerzos para determinar estándares en contenidos educativos que permitan su uso en distintos ambientes virtuales de aprendizaje. Entre estas especificaciones se encuentra *SCORM* (Sharable Content Object Reference Model), siendo uno de los más utilizados. Recientemente, la especificación *experience API* (también conocida como xAPI), mediante el registro de acciones educativas específicas, permite intercambiar experiencias de aprendizaje entre herramientas educativas digitales. Las características técnicas y funcionales de xAPI permiten atender los requerimientos operativos y de integración requeridos en el entorno educativo implementado que sustenta el presente estudio; ambiente educativo en línea derivado del proyecto RITADA. A continuación, se describe: 1) la especificación xAPI como protocolo de comunicación, así como 2) los mecanismos de integración al ambiente educativo implementado en el proyecto RITADA.

3.2.1 Especificación xAPI

Con la necesidad de establecer estándares que permitan compartir datos entre aplicaciones surgen propuestas como la ya comentada SCORM, las cuales buscan integrar funcionalidades y contenidos entre dichas aplicaciones. SCORM surge como un conjunto de estándares y especificaciones ante la necesidad de compartir contenidos educativos (objetos pedagógicos), disponibles en diversos sistemas de gestión de contenidos; implementados mediante formatos propietarios. Siendo el objetivo principal de la especificación permitir el intercambio entre dichos contenidos.

La primera versión de SCORM ADL (Advanced Distributed Learning), liberada en enero del año 2000, fue considerada como un borrador; ya que no contenía una especificación en sí, sino era solo una vista preliminar de lo que venía. SCORM 1.0 contenía los elementos principales que se convertirían en la base de SCORM, en particular mostraba la especificación de: 1) cómo se debe de empaquetar el contenido (empaquetado de contenido); 2) cómo debe de comunicarse el contenido con un LMS (tiempo de ejecución); y 3) cómo debe describirse el contenido (metadatos) [60].

Las versiones 1.1 y 1.2 de SCORM fueron liberadas en enero y octubre de 2001. Siendo las primeras versiones implementables de SCORM, todavía existen implementaciones funcionando bajo estos esquemas. La versión 1.2 fue considerada una especificación robusta e implementable, ya que los desarrolladores de contenidos

educativos se dieron cuenta de los ahorros que generaba la implementación gracias a la operatividad de contenidos; en la actualidad SCORM 1.2 es considerado el caballo de batalla de la industria [61]. Actualmente, se cuenta con SCORM 3ra y 4ta edición que permiten una mejor operatividad, la adición de especificaciones para la interfaz de usuario para los LMS y la creación de contenido secuenciado se estima sea más simple [48]. Recientemente, el integrar o concentrar datos provenientes de aplicaciones diferentes es permitido gracias a la creación de la especificación *Experience API* (xAPI), especificación impulsada por SCORM [62].

El estándar propuesto, *Tin Can API* o *xAPI*, es una especificación que describe las reglas de almacenamiento para el seguimiento de las experiencias o interacciones de aprendizaje de una persona. xAPI surge debido a las limitaciones que existían para compartir datos de una aplicación a otra con distinto lenguaje de programación. Se le conoce como sucesor de SCORM, y fue desarrollado por la *Iniciativa de Aprendizaje Distribuido Avanzado*, pero a diferencia de xAPI, SCORM no tiene un estándar abierto [63].

El gobierno de los Estados Unidos, mediante ADL, impulsó el desarrollo de Experience API. La primera versión de xAPI fue lanzada el 26 de abril de 2013, siendo considerado como la versión más nueva o complemento de SCORM y resuelve muchos problemas de sus versiones anteriores; tales como el aprendizaje móvil, el aprendizaje en equipo, la funcionalidad de secuenciación, no dependencia del ambiente web, entre otras. En la actualidad, xAPI ha sido adoptada por más de 200 organizaciones en E.U., un ejemplo es el departamento de la defensa de este país. En la Figura 3.4 se muestra cómo ha ido evolucionando la especificación Tin Can API, o xAPI [57].

El protocolo de xAPI que permite que dos o más entidades intercambien información es enviado a través de declaraciones tipo ontologías en la forma *actor*, *verbo* y *objeto*, el cual es almacenado en un repositorio común; denominado como *almacén de registros de aprendizaje* (LRS, por sus siglas en inglés) (ver Figura 3.5) [58]. En este formato el *actor* es el agente de la declaración, puede ser un alumno, un profesor o un grupo. El *verbo* describe la acción realizada, ejemplo: lectura, visualización o responder. El *objeto* es el elemento de aprendizaje con el que el actor interactúa, esta sintaxis genera una sentencia que se almacena mediante un registro en formato JSON (JavaScript Object Notation) en el LRS.

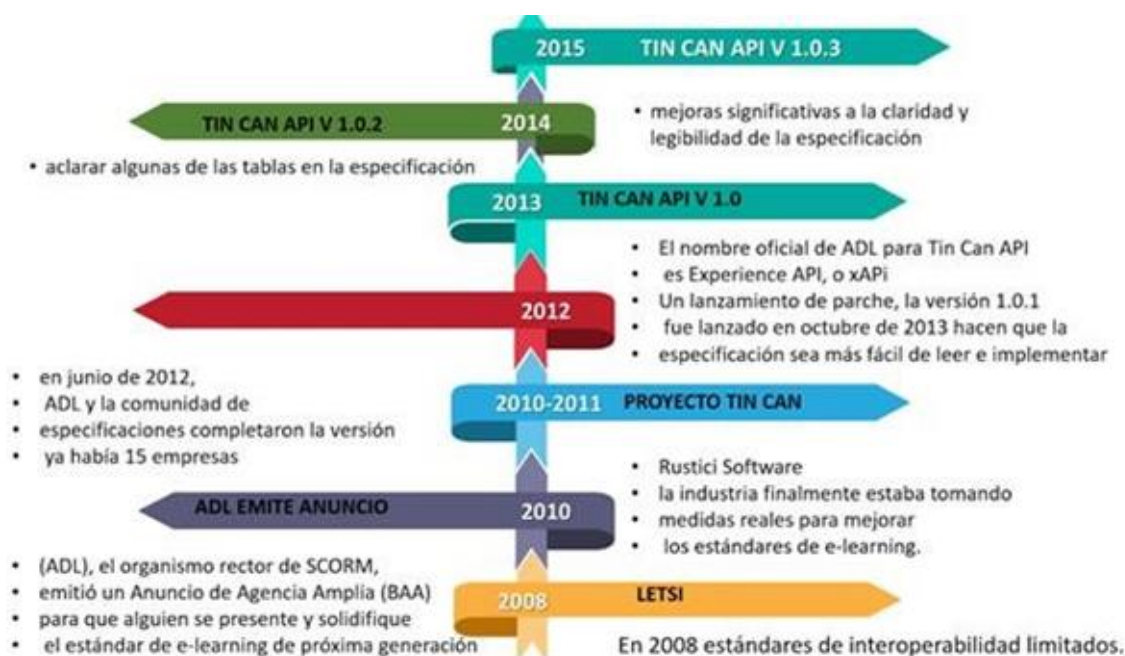


Figura 3.4. Evolución de especificación TIN CAN API (xAPI)

(Elaboración propia: datos provenientes de [57])

El protocolo de xAPI que permite que dos o más entidades intercambien información es enviado a través de declaraciones tipo ontologías en la forma *actor*, *verbo* y *objeto*, el cual es almacenado en un repositorio común; denominado como *almacén de registros de aprendizaje* (LRS, por sus siglas en inglés) (ver Figura 3.5) [58]. En este formato el *actor* es el agente de la declaración, puede ser un alumno, un profesor o un grupo. El *verbo* describe la acción realizada, ejemplo: lectura, visualización o responder. El *objeto* es el elemento de aprendizaje con el que el actor interactúa, esta sintaxis genera una sentencia que se almacena mediante un registro en formato JSON (JavaScript Object Notation) en el LRS.



Figura 3.5. Estructura de declaraciones de LRS (Elaboración propia en [58])

Los siguientes son algunos ejemplos de cada uno de los elementos principales de la especificación:

Ejemplo 1.

```

Actor
{
  "name": "Alan Martínez",
  "inbox": "mailto:rodolfo.martinez@uabc.edu.mx"
}

```

En este ejemplo se tienen 2 campos dentro de la etiqueta *Actor*, el campo *nombre* y el campo *inbox* (correo electrónico); considerando que el mismo nombre puede existir para personas diferentes, pero el correo sólo existirá para una persona; frecuentemente es utilizado como identificador único o principal [64].

Los verbos en Tin Can son palabras que describen lo que sucede entre el actor y un objeto, también se pueden representar como URIs (Identificador Uniforme de Recurso). En el ejemplo 2 indicamos que una persona ha aceptado una tarea [64].

Ejemplo 2

Verbo

```
{
  "name": {
    "en-US": "accepted"
  },
  "description": {
    "en-US": "indica que el actor ha aceptado el objeto", para esta
    instancia la persona aceptó una tarea.
  }
}
```

Para terminar, una sentencia o enunciado se describe el elemento objeto, siendo otra actividad que se identifica por su URI, ya que generalmente refiere a algún tipo de instrucción. En el ejemplo 3 se muestra la sintaxis del campo objeto [64].

Ejemplo 3

Objeto

```
{
  "id": "http://www.example.com/tincan/activities/baywxhelp",
  "objectType": "Actividad",
  "definition": {},
  "name": { },
  "description": {
    "en-US": "Actividad Ejemplo"
    "en-US": "Definición actividad ejemplo"
  }
}
```

Existen más campos que se pueden agregar a la estructura de un enunciado Tin Can API, como el contexto que ofrece un lugar; permitiendo añadir información sobre un enunciado. Por ejemplo: se puede agregar más información sobre la actividad, el

docente encargado del curso, la duración de la misma, entre otros. También se cuenta con el campo *resultado* donde es posible guardar información como la respuesta dada por el actor, si fue correcta o no, así como el número de intentos de solución de un ejercicio.

Una de las grandes aportaciones o intenciones de xAPI, es su premisa sobre que el aprendizaje se da en cualquier lugar donde interactúa el aprendiz. Con esto como base, la especificación busca que toda experiencia de aprendizaje sea registrada y consumida mediante el uso de un LRS; independientemente del contexto donde se genere: virtual, físico o híbrido. Tal como lo describen en su imagen promocional presentada en la Figura 3.6.



Figura 3.6. Estructura de declaraciones de LRS (Elaboración xAPI website [62])

Como se comentó anteriormente, la idea principal de xAPI es registrar experiencias de aprendizaje utilizando un formato estándar, las cuales puedan ser re-utilizadas o consumidas por la misma aplicación que las generó u otras distintas a ella. Dichas experiencias se registran en uno o múltiples LRS, tal como se describe a continuación.

3.2.2 ¿Qué es un LRS (Learning Record Store)?

Un *Contenedor de Registros de Aprendizaje* (LRS por sus siglas en inglés) es un sistema que permite almacenar y proveer experiencias de aprendizaje. La información es generada como resultado de las experiencias o interacciones realizadas en otras aplicaciones o dispositivos, con la única restricción de que cumplan con la

especificación xAPI. Sin embargo, el formato para grabar las experiencias de aprendizaje es demasiado flexible, debido a que las experiencias se registran en un formato de verbo, sujeto y actividad [65]

¿Para qué sirve un LRS?

Un LRS recopila, almacena, ordena y comparte los datos provenientes de distintas aplicaciones (virtuales o físicas), obligando a que la estructura de los datos sea homogénea; de forma que el lenguaje sea el mismo para todas las aplicaciones a través del estándar xAPI [65].

Diferencia entre LRS y LMS

Aun cuando el enfoque de LRS y LMS es gestionar elementos de aprendizaje, no es lo mismo un LRS que un LMS; ni se considera que lo reemplazará. El LRS complementa los entornos de aprendizaje, ya que puede observar y registrar que es lo que sucede en el exterior de un LMS y no solo se limita a registrar lo que sucede en el interior de estas plataformas.

Los servicios de un LRS pueden estar ligados a los que brinda un LMS, el LRS puede utilizar los datos que se obtienen mediante el LMS para realizar reportes y análisis. Mediante un LRS se puede llevar el registro del rendimiento de los alumnos en un entorno escolar, los resultados nos pueden ayudar a implementar estrategias de enseñanza más acordes a las deficiencias encontrada [66].

Sistemas que nos permiten implementar un LRS

- Yet analytics,
- Watershed LRS,
- Wax LRS,
- Learning locker,
- ADL LRS para moodle,
- Plugin para moodle:

Tin Can API launch link (para moodle)

Logstore API (para moodle)

En resumen, los LRS pueden funcionar solos o como parte de un LMS, siendo una nueva especificación para las aplicaciones de *e-learning* que permiten almacenar las interacciones de una persona ante distintas actividades, ya sea online y/o offline. El uso de esta API permite que sistemas muy diferentes puedan comunicarse entre sí mientras interactúan con diferentes tecnologías y contenidos [62]. Con base en las bondades ofrecidas por la especificación xAPI se tomó la decisión de utilizarla para responder nuestra tercera pregunta de investigación *¿Qué protocolo de comunicación entre tutores inteligentes, basado en la tecnología multiagentes permite la interacción entre ellos?*

3.3 Evaluación del desempeño de la arquitectura de software propuesta al interactuar con múltiples tutores inteligentes provenientes de diversos dominios

Para responder a nuestra cuarta pregunta de investigación *¿Cómo lograr un desempeño operativo de la arquitectura propuesta al interactuar con múltiples tutores inteligentes de diversos dominios, considerando sus características de operatividad, rendimiento, compatibilidad, y escalabilidad?*, se llevó a cabo una serie de tres evaluaciones de forma secuencial, considerando las pruebas de calidad de software con base en los criterios de operatividad, rendimiento, compatibilidad y escalabilidad; definidas en el modelo de calidad ISO/IEC 9126-1. A continuación se presenta el método utilizado en esta fase de la investigación, los participantes en el estudio, los instrumentos y técnicas de recolección de información, y el proceso de análisis de los datos. Los resultados de esta fase del estudio se presentan en el capítulo 4 de esta tesis.

Para realizar la evaluación del desempeño de la arquitectura implementada se utilizó una metodología cuantitativa, lo anterior considerando que este estudio se basó en aspectos observables y medibles en nuestro entorno real [67]. Particularmente, se procedió a medir la relación existente entre las características del sistema al interactuar con múltiples tutores inteligentes que atienden dominios diversos: operatividad, rendimiento, compatibilidad y escalabilidad. Las tareas realizadas en esta parte de la investigación se detallan a continuación.

Obtención de permisos de intervención.

Se procedió a promover el uso del sistema integral de tutores inteligentes en facultades y escuelas de la Universidad Autónoma de Baja California y externas. Se obtuvo la aceptación de uso del ambiente en dos instituciones externas a UABC, CETyS Universidad y CETMAR (nivel medio superior), y dos Facultades de UABC campus Ensenada (Facultad de Ciencias y Facultad de Ciencias Administrativas y Sociales). Finalmente, el estudio se focalizó con los estudiantes de nuevo ingreso de la Facultad de Ciencias Administrativas y Sociales, esto debido a la cantidad de estudiantes que ingresan a esta unidad académica y la diversidad de carreras a la que se dirigen los estudiantes una vez concluidos sus créditos de tronco común.

Identificación de la muestra.

Considerando la factibilidad y apertura de los directivos y docentes de la Facultad de Ciencias Administrativas y Sociales, se tomó la decisión de llevar a cabo la evaluación del uso del sistema con los 10 grupos que ingresaron en el periodo 2018-1 en el área económica administrativa. El estudio realizado por alrededor de 400 alumnos, consistiendo del 100% del universo de ingreso, permitió llevar a cabo las pruebas de calidad de software consideradas en la investigación: operatividad, rendimiento, compatibilidad y escalabilidad.

Recolección de datos.

La totalidad de los datos se obtuvieron de los mecanismos de registro automatizados con los que cuenta el ambiente, los cuales consisten de los siguientes registros:

- a. ingreso al sistema,
- b. interacciones con los materiales educativos por estudiante y grupal, y
- c. asesorías brindadas por los tutores inteligentes.

Lo anterior permitió cuantificar la cantidad y diversidad de información procesada por el ambiente. Adicionalmente, se procedió a evaluar los tiempos de respuesta de la aplicación registrados en los archivos del sistema operativo del servidor que hospeda los servicios de tutoría inteligente brindados por el ambiente desarrollado.

Análisis de datos cuantitativos.

El análisis de los datos obtenidos, primeramente, se realizó mediante un análisis de estadísticas descriptivas. Lo anterior con el objetivo de identificar comportamientos en el desempeño del ambiente propuesto. Se identificaron aspectos esenciales como resultados significativos, resultados no significativos, datos atípicos y diferencia entre grupos.

Discusión y conclusiones del estudio.

Se concluye esta fase del estudio con el análisis detallado de los resultados obtenidos y una discusión de lo encontrado. Buscando dar explicación a lo observado e identificar el nivel de generalización al que se puede llegar después del estudio. Lo anterior lleva a poder determinar conclusiones del estudio, líneas de investigación factibles de explorar y trabajo futuro del grupo de investigadores participantes en el estudio.

3.4 Evaluación de la aceptación y eficiencia de la tutoría brindada

Para responder a nuestra quinta pregunta de investigación *¿Qué nivel de aceptación y eficiencia se logra al brindar asesoría, mediante un ambiente interactivo de tutores inteligentes que estandariza y comparte la base de conocimiento del estudiante?*, se llevaron a cabo actividades divididas en dos fases realizadas de forma secuencial. La primera fase estuvo orientada a realizar el diagnóstico de los estudiantes participantes y alimentar la base de conocimiento conductual del tutor inteligente, y la segunda para medir la satisfacción y aceptación de la eficiencia de la tutoría personalizada proporcionada. A continuación, se presenta el método utilizado en esta parte de la investigación, los participantes en el estudio, los instrumentos y técnicas de recolección de información, así como el proceso de análisis de los datos. Los resultados de esta fase del estudio se presentan en el capítulo 5 de esta tesis.

Para la evaluación de satisfacción y aceptación de la arquitectura implementada se utilizó una metodología cuantitativa. Particularmente, se midió la relación existente

entre estilo de aprendizaje y autoestima con la aceptación y eficiencia de la asesoría brindada. Las tareas realizadas en esta parte de la investigación se detallan a continuación.

Obtención de permisos de intervención.

Una vez dado a conocer el uso y características del sistema integral de tutores inteligentes, en la Facultad de Ciencias Administrativas y Sociales de la Universidad Autónoma de Baja California, se procedió a gestionar los permisos con directivos, profesores y estudiantes de los grupos participantes en esta etapa del estudio.

Identificación de la muestra.

Considerando la factibilidad y apertura de los directivos y docentes invitados a participar, mostrada en la etapa previa de esta investigación, se tomó la decisión de llevar a cabo la evaluación del uso del sistema con los 4 grupos que ingresaron al área económica administrativa de la unidad de aprendizaje Comunicación Oral y Escrita en el periodo 2019-2. El estudio realizado con los 4 grupos, que tienen en su plan de estudio la unidad de aprendizaje Comunicación Oral y Escrita, consistió en la obtención de su nivel de autoestima, donde participaron 141 alumnos. Para la obtención del estilo de aprendizaje se contó con la participación de 114 estudiantes de los mismos grupos, siendo el 50% del universo de grupos de nuevo ingreso de la Facultad participante.

Recolección de datos.

La totalidad de los datos en la primera fase de esta evaluación se obtuvieron de los mecanismos de registro automatizados con los que cuenta el ambiente, los cuales consisten de los siguientes registros:

- a. Instrumento diagnóstico de estilo de aprendizaje,
- b. Instrumento diagnóstico de autoestima.

Lo anterior permitió cuantificar la cantidad y diversidad de información procesada por el ambiente, dividiendo a los participantes por estilo de aprendizaje y nivel de autoestima. Para la segunda fase de esta evaluación se procedió a solicitar a los estudiantes participantes que resolvieran el ejercicio de Clasificación de Palabras (agudas, graves y esdrújulas), perteneciente a la sección de comunicación oral y escrita. El

ejercicio fue seleccionado por ser considerado un ejercicio con una complejidad media-baja. Posteriormente, se procedió a entregar una encuesta para medir la aceptación y la eficiencia de la tutoría personalizada brindada. La encuesta fue adaptada del instrumento de usabilidad y experiencia de usuario UMUX-Lite [68], y validada por un experto en psicología pedagógica.

Análisis de datos cuantitativos.

El análisis de los datos obtenidos en la encuesta contestada se llevó a cabo mediante un análisis de estadísticas descriptivas. Lo anterior con el objetivo de medir la aceptación y eficiencia de la asesoría personalizada brindada por el ambiente propuesto. Se identificaron aspectos esenciales como resultados significativos, resultados no significativos, datos atípicos y diferencia entre grupos por su estilo de aprendizaje y nivel de autoestima.

Discusión y conclusiones del estudio.

Se concluye esta fase del estudio con el análisis detallado de los resultados obtenidos y una discusión de lo encontrado. Buscando explicar el grado de aceptación y eficiencia de la asesoría personalizada recibida durante el ejercicio por parte de los estudiantes de los grupos evaluados. Lo anterior permitió determinar conclusiones del estudio, líneas de investigación factibles de explorar y trabajo futuro del grupo de investigadores participantes en el estudio.

Capítulo 4

Diseño y desarrollo del ambiente integral de aprendizaje

En este capítulo se describe el proceso de implementación del ambiente integral de tutores inteligentes propuesto, incluyendo el sustento tecnológico soportado por la tecnología *experince API*. Adicionalmente, se presentan los resultados obtenidos en una primera fase de evaluación del ambiente.

4.1. Integración de xAPI en el proyecto RITADA

Para utilizar la especificación xAPI es necesario contratar los servicios de un LRS, o instalar y configurar uno propio. Las ventajas de la segunda opción es que permite eliminar dependencias de terceros, se tiene el control directo de los datos generados, sobre todo si algunos de estos registros son de carácter confidencial, y evita gastos significativos. Sin embargo, se enfrenta el reto de instalar y configurar dicho servicio, que dada su reciente aparición existe poca documentación o fuentes de asesoría al respecto. No obstante, considerando el volumen de transacciones que se estima generará el ambiente compartido implementado, se procedió con la instalación de un LRS propietario.

Se determinó utilizar el LRS *Learning Locker*. Sistema seleccionado por ser de los más utilizados y recomendado en la comunidad usuaria de xAPI, contar con una librería estable (plug-in) que permite conectar con LMS como Moodle (Tin Can Launch), ser un LRS de código abierto (open source), y contar con un elaborado módulo de analítica.

Una vez que se cuenta con un LRS, es importante configurar los servicios de enseñanza-aprendizaje que harán uso de este. En el caso particular de esta investigación, se instaló y configuró el módulo Tin Can Launch en Moodle, LMS base utilizado en el ambiente federado. Así mismo, se procedió a utilizar la librería *tincan.js*, la cual nos permitió comunicar aplicaciones propietarias con el LRS (ej. módulos de tutores inteligentes).

En la Figura 4.1 se describe un ejemplo de la arquitectura de un ambiente educativo que soporta la especificación xAPI. El ejemplo describe el uso de xAPI aplicado a un ambiente cliente-servidor, sin embargo, recordemos que este puede ser extendido a aplicaciones sin conectividad a internet (off-line), a través de actividades de aprendizaje en entornos físicos que puedan ser monitoreadas y registradas mediante otros mecanismos (sensores).

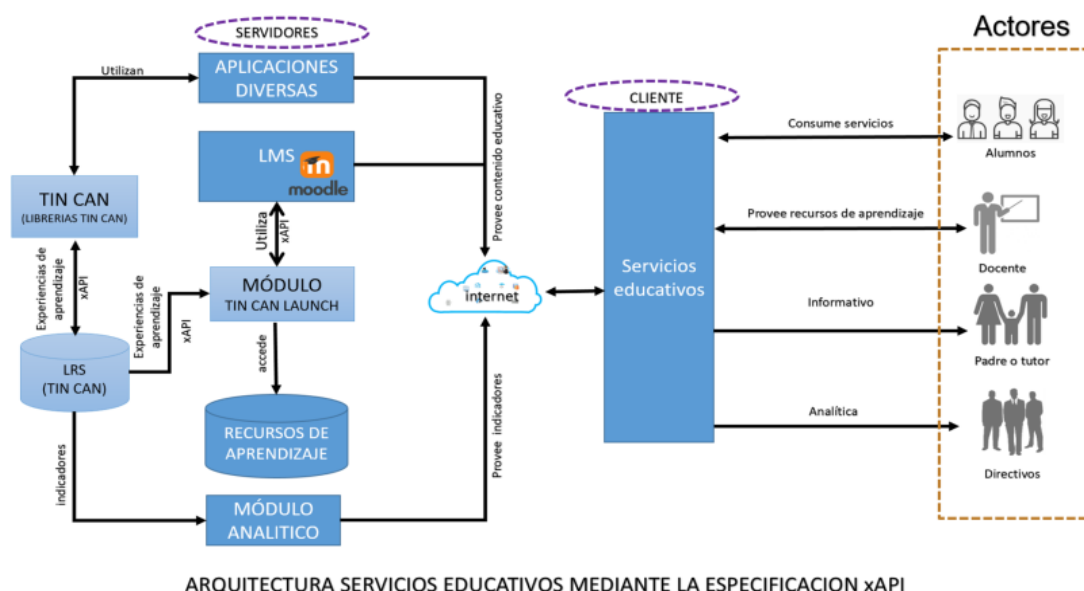


Figura 4.1. Arquitectura de servicios educativos mediante la especificación xAPI
(Elaboración propia)

Como es evidente, la arquitectura de xAPI descrita en la Figura 4.1 refiere a una arquitectura cliente-servidor, ambiente donde el cliente, a través de un navegador, demanda realizar una tarea o servicio proveído por el servidor (ej. actualizar datos del estudiante en la base de datos del ambiente educativo). Por el lado del cliente vemos a los roles o actores que interactúan en el ambiente (alumnos, docentes, padres o tutores y directivos).

Específicamente, los alumnos consultan el material educativo a través de la red, e ingresan a la aplicación Web mediante autenticación. Al momento de realizar las actividades se genera un registro en el LRS con los detalles de cada actividad llevada a cabo; permitiendo el monitoreo de sus avances, tanto a nivel cognitivo como aspectos de tipo conductual. Por otro lado, los docentes se encargan de alimentar el ambiente con recursos educativos, ya sea elaborando material o estableciendo actividades para los

alumnos, además de monitorear los registros referentes a las actividades realizadas por los alumnos y sus avances.

Es posible definir usuarios con figura de observador. Por ejemplo, los padres o tutores solo pueden fungir como observadores, ellos pueden ingresar a través de internet previo un proceso de autenticación, para monitorear el desempeño de sus hijos. Los directivos, al igual que los padres o tutores, solo tienen un rol de observadores, debido a que solo pueden revisar indicadores, ya sea para el monitoreo del desempeño de sus docentes o alumnos.

El LRS es el encargado de almacenar la información relacionada con las experiencias de aprendizaje de los alumnos. El módulo Tin Can Launch permite la conexión e intercambio de información entre el LMS (Moodle, en el caso particular de este estudio) y el LRS. Para que otras aplicaciones se puedan conectar con el LRS se debe agregar una librería específica de Tin Can (ej. tincan.js). El LRS también permite acceder a su módulo de analítica, todo esto a través de internet

En la Figura 4.2 se presenta la implementación de la especificación en el ambiente federado desarrollado para el proyecto RITADA. Se cuenta con la red del ambiente federado donde se comunican los servidores de UCOL, SIEC y RITADA, pertenecientes a distintas unidades de educación superior. En el proyecto RITADA, UABC y UCOL contienen los servicios donde se gestionan datos, permitiendo que la compartición sea posible al iniciar sesión en la base de datos de un servidor de autenticación de la zona a la que pertenece el usuario; pero gracias al sistema federado puede acceder a los servicios ofrecidos por cualquiera de las otras instituciones participantes en la red. Cada servidor tiene su base de datos, pero para centralizar la información relevante o genérica de cada estudiante se utilizó un LRS; en este caso Learning Locker. Lo anterior, permite la comunicación entre sistemas sin importar el lenguaje de programación que se utilice en cada institución.

Al momento, se cuenta con la integración de la especificación en los siguientes servicios educativos: *PreMath* que contiene ejercicios de matemáticas preuniversitarias, *COE* que incluye ejercicios para Comunicación Oral y Escrita, *Autoestima* que contiene videos interactivos de Autoestima, y *Hábitos de Estudio* que es un instrumento que obtiene los hábitos de estudio del participante en el ambiente RITADA. Dicha

integración se ha llevado a cabo mediante el registro de experiencias de aprendizaje (_set) y el consumo de dichos registros (_get) para la personalización de la asesoría. En la Figura 4.3 se presenta un historial de registros derivado de su funcionalidad.

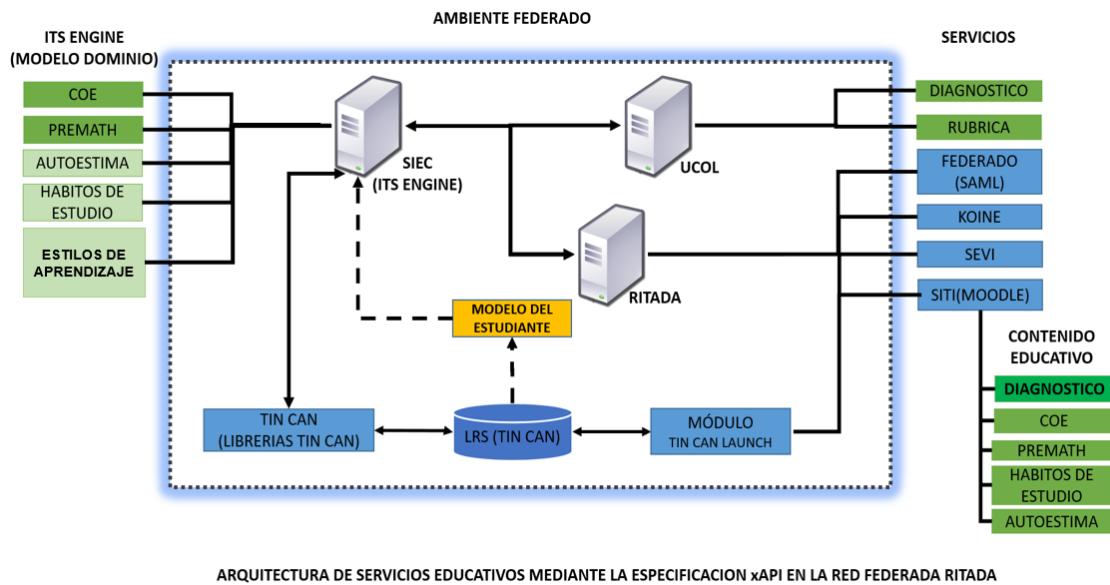


Figura 4.2. Arquitectura de servicios educativos mediante la especificación xAPI en la red federada RITADA (Elaboración propia).

>	Ángel Alan Castañeda	viewed Interpretación de imágenes	3 hours ago
>	Ángel Alan Castañeda	viewed Interpretación de imágenes	3 hours ago
>	Ángel Alan Castañeda	viewed Interpretación de imágenes	3 hours ago
>	Ángel Alan Castañeda	viewed Comunicación Oral y Escrita	3 hours ago
>	Ángel Alan Castañeda	viewed Comunicación Oral y Escrita	3 hours ago
>	Ángel Alan Castañeda	viewed Comunicación Oral y Escrita	3 hours ago
>	Ángel Alan Castañeda	viewed Comunicación Oral y Escrita	3 hours ago
>	Ángel Alan Castañeda	viewed Comunicación Oral y Escrita	3 hours ago
>	Ángel Alan Castañeda	viewed TDA	3 hours ago
>	Ángel Alan Castañeda	viewed Autoestima	3 hours ago
>	Ángel Alan Castañeda	viewed TDA	3 hours ago
>	Ángel Alan Castañeda	viewed TDA	3 hours ago
>	Ángel Alan Castañeda	viewed TDA	3 hours ago

Figura 4.3. Historial de registros derivado de la funcionalidad de un LRS (Elaboración propia).

4.2. Evaluación preliminar del ambiente implementado

A continuación, se describen los resultados de la evaluación realizada en el entorno de aprendizaje implementado. El objetivo de esta evaluación es determinar el desempeño de la plataforma RITADA posterior a la integración del sistema LRS y de la generación de datos para el modelo del estudiante implementado. Específicamente, se evaluó su nivel de *operatividad, rendimiento, compatibilidad y escalabilidad*; características objetivo de nuestra investigación, las cuales deben ser medidas en un ambiente con un volumen alto de usuarios concurrentes utilizando distintos tutores inteligentes. Esta evaluación se realizó con la participación de usuarios reales de nivel universitario, el diseño del estudio y los resultados del mismo se describen a continuación.

4.2.1 Participantes

Los participantes en el estudio preliminar fueron alrededor de 400 alumnos pertenecientes a 10 grupos de la *Facultad de Ciencias Administrativas y Sociales* del tronco común de la *Universidad Autónoma de Baja California*, que ingresaron al área económica administrativa. El total de los participantes correspondió al 100% de los grupos de nuevo ingreso del periodo 2018-1.

4.2.2 Instrumentos

Los instrumentos empleados en la recolección de los datos fueron automatizados. Primeramente, se utilizó el módulo de diagnóstico de la plataforma educativa RITADA para obtener el nivel de ingreso de los alumnos participantes, la plataforma proporciona cinco fuentes de información diagnóstica, las cuales son:

- 1) Conocimiento en matemáticas.
- 2) Hábitos de estudio.
- 3) Autoestima.
- 4) Comunicación oral y escrita.
- 5) Estilos de aprendizaje.

Cada encuesta utilizada para la recolección de datos se derivó de instrumentos validados y disponibles en la literatura. Adicionalmente, los instrumentos generados se evaluaron con el apoyo de expertos en el área [69].

Los tutores inteligentes utilizados corresponden a los dominios de matemáticas preuniversitarias, hábitos de estudio y autoestima. El ITS de matemáticas incluye ejercicios de 5 temáticas distintas, el registro en el LRS que se tiene al momento corresponde al acceso de los ejercicios. Los dominios de hábitos de estudio y autoestima consisten de vídeos interactivos, de los cuales se registra el acceso y las respuestas obtenidas por los estudiantes. Los vídeos interactivos, implementados con el software H5P, consisten de preguntas embebidas que se despliegan al reproducir el material.

4.2.3 Procedimiento

El estudio preliminar se realizó en un periodo de dos semanas utilizando los diferentes módulos de ITS de RITADA, donde se obtuvo información cuantitativa sobre el desempeño de los estudiantes y su nivel de interacción con la plataforma. Para la obtención de los datos cuantitativos del experimento se realizaron las siguientes actividades:

1. Capacitación a estudiantes en el uso y registro de la plataforma educativa RITADA.
2. Acceso a la plataforma RITADA.
3. Ejecución y monitoreo constante del experimento.
4. Obtención de resultados.
5. Análisis de los datos Obtenidos.
6. Elaboración de conclusiones.

4.2.4 Técnicas de análisis y evaluación preliminar

El análisis realizado al sistema implementado se basa en medir el rendimiento del ambiente educativo con base en métricas de desempeño de software. Las métricas de desempeño de un sistema basado en cómputo miden el rendimiento de módulos y sistemas

de software completos; particularmente se consideraron las siguientes tres métricas de desempeño de software definidas por el modelo de calidad ISO/IEC 9126 1, y una adicional considerada relevante por el tipo de implementación realizada: operatividad (funcionalidad), rendimiento (eficiencia), compatibilidad (portabilidad) y escalabilidad (adicional):

- **Operatividad**, capacidad de intercambio de información entre distintos sistemas.
- **Rendimiento**, tiempo de respuesta y uso adecuado de recursos.
- **Compatibilidad**, integración y capacidad de coexistencia.
- **Escalabilidad**, capacidad de manipular múltiples tutores inteligentes.

Operatividad

La medición del nivel de operatividad del ambiente propuesto se ha realizado considerando dos aspectos principales: (1) capacidad de compartición de información entre el modelo del dominio y el modelo del estudiante, a través del protocolo utilizado por el LRS; y (2) capacidad de uso de información entre múltiples modelos del dominio.

La plataforma RITADA cuenta con contenido instruccional de 4 dominios distintos, 2 dentro de la categoría conductual y 2 en la categoría de conocimientos: *matemáticas preuniversitarias, comunicación oral y escrita, hábitos de estudio y autoestima*. Al momento, se ha configurado el ambiente para que todos los módulos instruccionales utilicen el servicio de registro (_set) en el LRS de la plataforma. El módulo de matemáticas y de comunicación oral y escrita registran el acceso a cada una de los temas y subtemas (ver Figura 4.4). Los módulos de hábitos de estudio y autoestima registran acceso, respuestas y conclusión de la actividad (ver Figura 4.5).

> User-ID 479 accessed MAT2018 EXP-1	3 months ago
> User-ID 479 accessed MAT2018 EXP-1	3 months ago
> User-ID 479 accessed MAT2018 EXP-1	3 months ago
> User-ID 479 accessed MAT2018 EXP-1	3 months ago
> User-ID 479 accessed MAT2018 EXP-1	3 months ago
> User-ID 554 accessed MAT2018 EXP-1	3 months ago
> User-ID 475 accessed MAT2018 EXP-1	3 months ago
> User-ID 475 accessed MAT2018 EXP-1	3 months ago
> User-ID 2 accessed COE2018 HOM-1	3 months ago
> User-ID 2 accessed MAT2018 EXP-1	3 months ago
> User-ID 2 accessed MAT2018 EXP-1	3 months ago

Figura 4.4. Historial de acceso a los módulos de matemáticas y comunicación oral y escrita (Elaboración propia, proviene de sistema RITADA).

>	Sonia Gutierrez attempted Motivación	3 months ago
>	Rosalva Borraz completed Motivación	4 months ago
▼	Rosalva Borraz attempted Motivación	4 months ago

```
{
  "refs": [],
  "stored": "2018-02-11T04:04:55.811Z",
  "active": true,
  "completedForwardingQueue": [],
  "failedForwardingLog": [],
  "client": "5a273652a3ac2203b4d81bae",
  "lrs_id": "5a273652a3ac2203b4d81bad",
  "completedQueues": [
    "STATEMENT_QUERYBUILDERCACHE_QUEUE",
    "STATEMENT_FORWARDING_QUEUE",
    "STATEMENT_PERSON_QUEUE"
  ],
  "hash": "91845bee0875dee186dcf7fee21f493841751e6b",
  "statement": {
    "authority": {
      "mbox": "mailto:hello@learninglocker.net",
      "name": "New Client",
      "objectType": "Agent"
    },
    "stored": "2018-02-11T04:04:55.811Z",
```

Figura 4.5. Historial de acceso a los módulos de hábitos de estudio y autoestima (Elaboración propia, proviene de sistema RITADA).

Rendimiento

El rendimiento del sistema, específicamente considerado como la capacidad de procesamiento de información ante un conjunto de peticiones concurrentes, se midió en forma cuantitativa analizando el registro de acciones almacenadas en el LRS ejecutadas en un mismo periodo de tiempo.

El acceso concurrente del sistema se evaluó mediante sesiones grupales en donde se les solicitaba a los estudiantes hacer uso de los diferentes módulos de la plataforma. Este estudio se realizó con 11 grupos con un rango entre 30 y 40 estudiantes (ver Figura 4.6). Durante la interacción de los estudiantes con el sistema se registraba su acceso y seguimiento de actividades. Las gráficas de las Figuras 4.7 y 4.8 muestran un ejemplo del número de accesos concurrentes presentados en una de las sesiones grupales. Es importante mencionar que los estudiantes realizaron tareas en el sistema después las sesiones grupales, pero en este caso el número de peticiones en forma concurrente fue menor.



Figura 4.6. Imágenes del uso del sistema en forma concurrente (Elaboración propia).

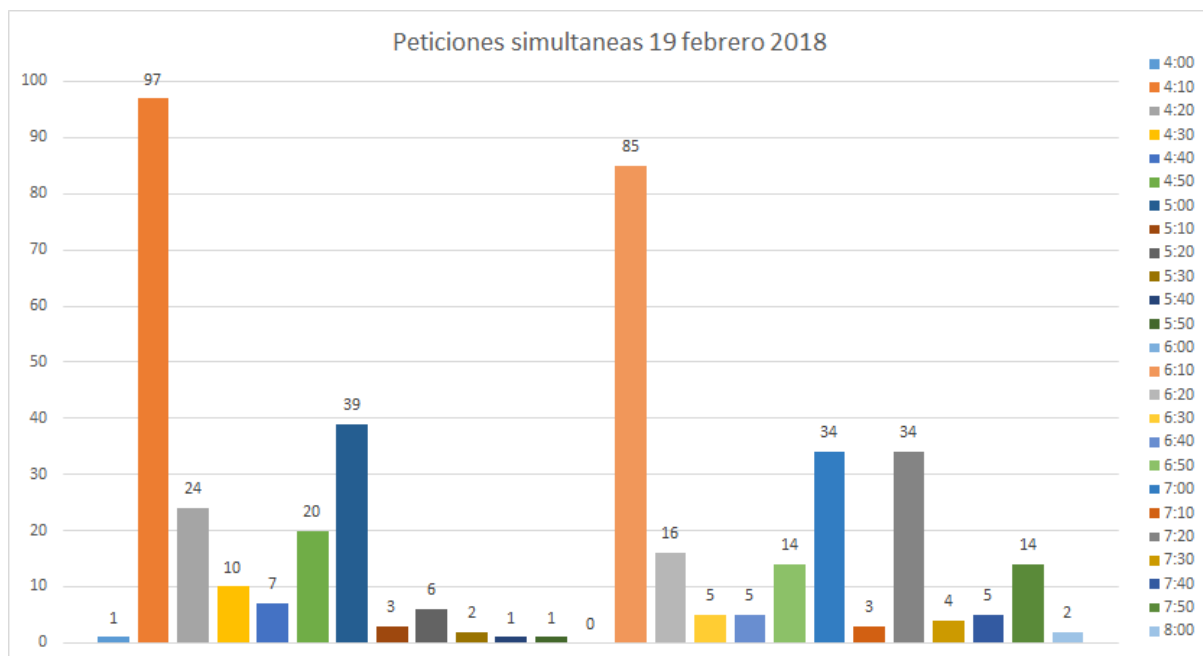


Figura 4.7. Accesos concurrentes mediante sesión grupal día 1 (Elaboración propia).

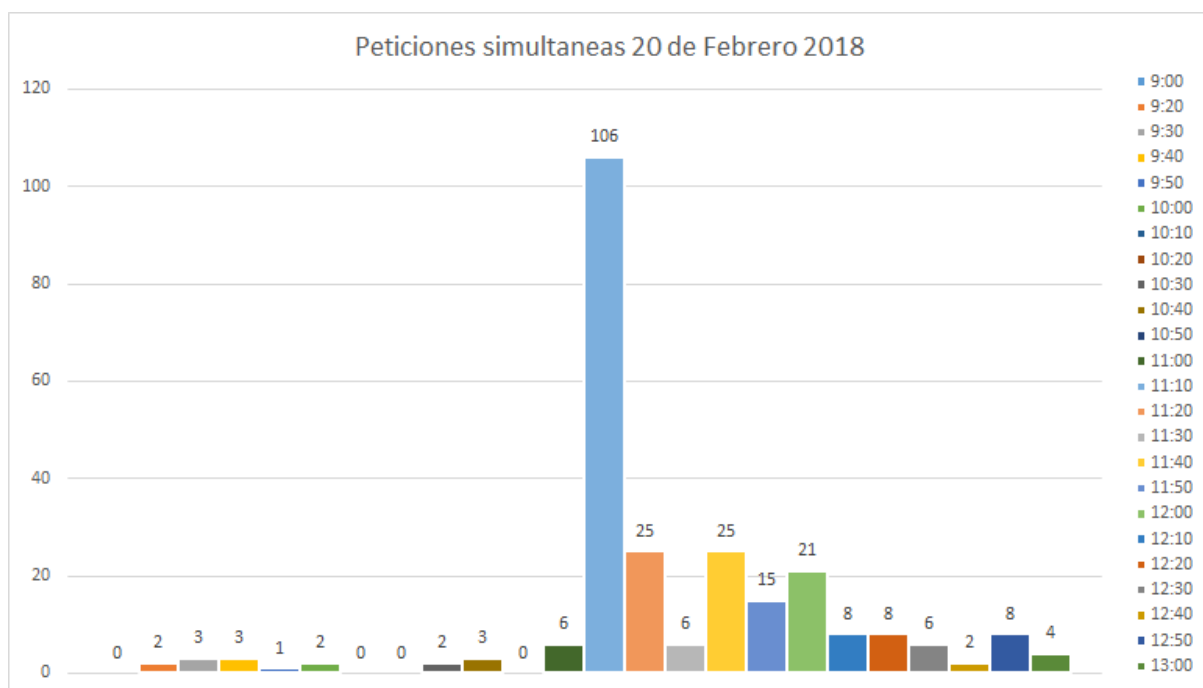


Figura 4.8. Accesos concurrentes mediante sesión grupal día 2 (Elaboración propia).

Compatibilidad

Mediante esta métrica se determina el nivel de integración con que cuenta el ambiente integral de tutores inteligentes al permitir la comunicación entre sistemas implementados con tecnologías heterogéneas.

Del contenido instruccional perteneciente a los 4 dominios que atiende actualmente la plataforma RITADA, los cuales proporcionan información al LRS, 2 de ellos fueron implementados mediante el software H5P (hábitos de estudio y autoestima), y la funcionalidad de los dos módulos restantes está basada en aplicaciones propietarias desarrolladas en PHP (matemáticas y comunicación oral y escrita). H5P es un software basado en HTML5 que permite crear contenido multimedia interactivo para Web, el cual consta de un complemento para interactuar con un LRS. Un ejemplo de esta implementación se presenta en la Figura 4.9, la cual incluye un vídeo interactivo con preguntas embebidas. La interacción de los módulos en PHP se realizó utilizando la librería en JavaScript *TinCanJS*, la cual consiste de métodos que permiten interactuar con la especificación Experience API. De igual forma, el gestor de contenidos usado en el ambiente RITADA (Moodle) registra en el LRS las entradas (login) y salidas (logout) al sistema mediante el complemento *Tin Can Launch*.

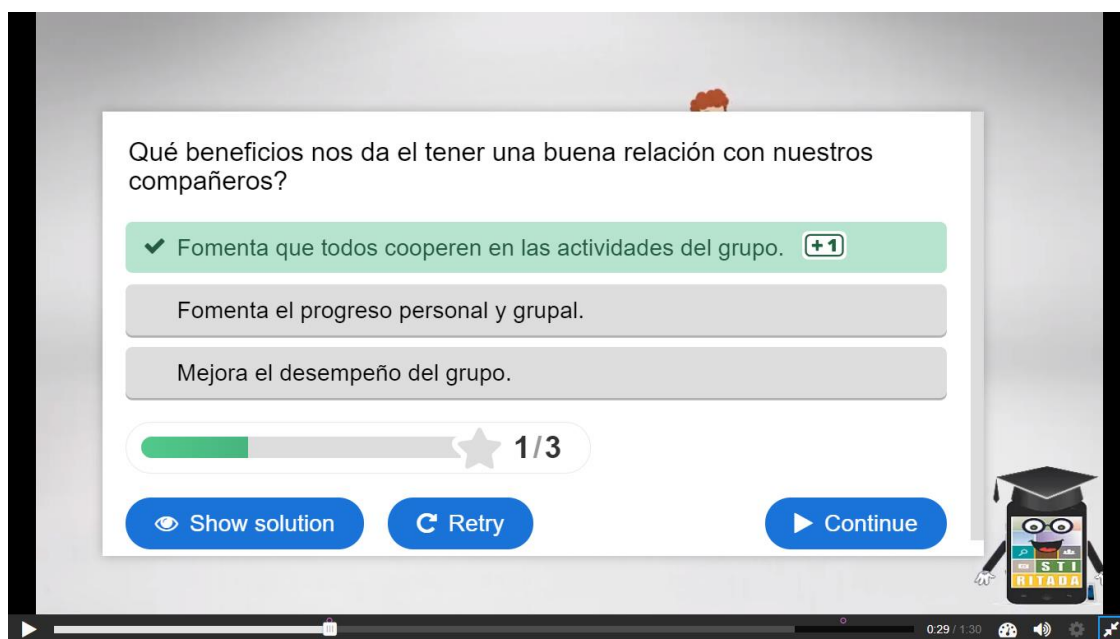


Figura 4.9. Implementación de H5P contenido multimedia interactivo para Web en RITADA
(Elaboración propia).

Con la implementación lograda al momento, y el desempeño obtenido al utilizar la plataforma RITADA en su conjunto, es evidente la compatibilidad existente entre sus componentes actuales (tutores inteligentes), el protocolo utilizado para el intercambio de información entre ellos (especificación xAPI), y el repositorio de actividades (LRS).

Escalabilidad

A este respecto, se evalúa la capacidad del sistema para integrar, comunicar y operar sistemas de tutoría inteligentes provenientes de dominios diversos. Al momento se tiene configurado en el ambiente la interacción entre 2 ITS orientados a fortalecer aspectos cognitivos y 2 dirigidos a características conductuales. Los ITS conductuales, constantemente actualizan la parte genérica del modelo del estudiante, mientras que los ITS cognitivos sólo influyen en los conocimientos específicos del dominio que atienden. La Figura 4.10 muestra las diferentes capas con que opera el modelo del estudiante propuesto para el ambiente RITADA.

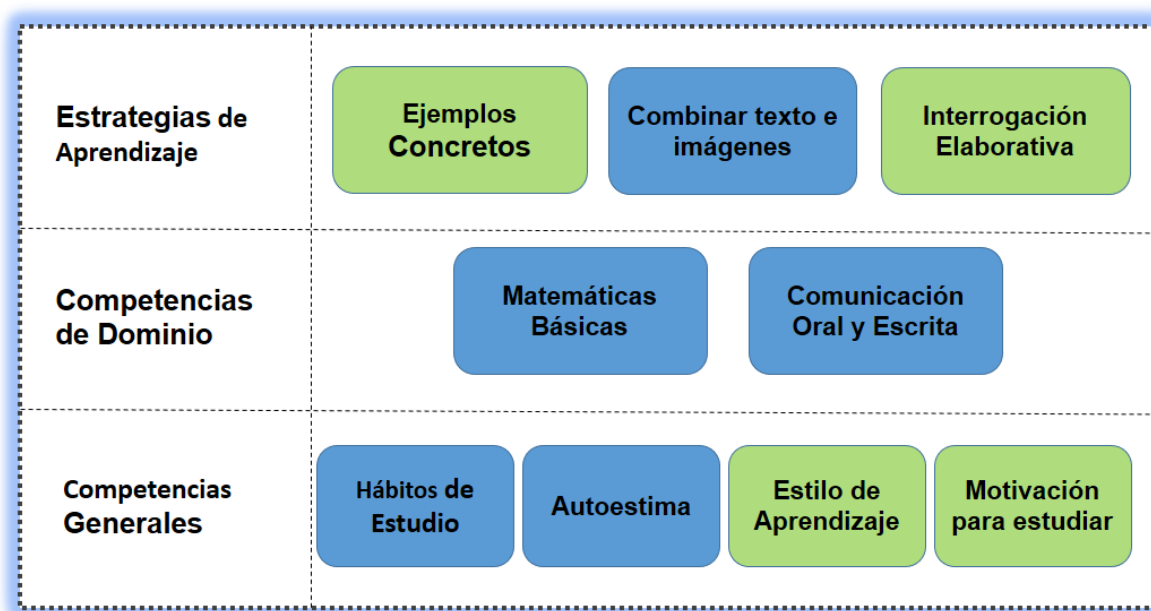


Figura 4.10. Capas del modelo del estudiante propuesto en RITADA
(Elaboración propia).

Se concluye esta fase del estudio con el análisis de los resultados obtenidos y una discusión de lo encontrado en el capítulo 6 de esta investigación.

Capítulo 5

Evaluación de satisfacción y aceptación de uso de la asesoría brindada por la arquitectura propuesta

Al concluir con el desarrollo y la implementación de cada uno de los módulos de la arquitectura RITADA se trabajó con las evaluaciones correspondientes a los módulos desarrollados en el ambiente propuesto. A partir de los resultados obtenidos en el estudio preliminar, orientados a medir el desempeño de la arquitectura propuesta, se determinó que la herramienta tiene un funcionamiento estable. Enseguida se llevó a cabo el estudio final, teniendo como objetivo la evaluación de la aceptación y satisfacción al recibir retroalimentación personalizada por la plataforma RITADA, empleando la arquitectura de software propuesta para la comunicación entre tutores inteligentes, basada principalmente en la homogeneización del modelo del estudiante.

En el presente capítulo se describen los resultados obtenidos en la evaluación de experiencia de usuario realizada a través del uso de la arquitectura de software propuesta. El objetivo de esta evaluación fue determinar el grado de satisfacción de usuarios a través de la retroalimentación recibida al interactuar con la plataforma RITADA; posterior a la integración de nuestra ontología y el LRS al entorno de aprendizaje. Específicamente, se evaluó el nivel de satisfacción y aceptación de la tutoría recibida al realizar el ejercicio de clasificación de palabras con base en su sílaba tónica, correspondiente al módulo de comunicación oral y escrita (ver Figura 5.1).

5.1 Participantes en el estudio de experiencia de usuario

Al igual que en la evaluación anterior, como parte de este estudio, se contó con la participación de 4 grupos del tronco común de la FCAyS, resultando con la participación de 141 alumnos en la etapa de obtención de nivel de autoestima y para la obtención del estilo de aprendizaje se contó con la participación de 114 alumnos de estos mismos grupos. El proceso se llevó a cabo durante un periodo de cuatro semanas. Los grupos que participaron en esta evaluación fueron distintos a los atendidos en el estudio de desempeño de la arquitectura descrito en el capítulo 4. Entre los participantes se contó

con una distribución homogénea de 72 mujeres y 62 hombres para la etapa de obtención de nivel de autoestima y para la fase de obtención de estilo de aprendizaje 59 mujeres y 55 hombres, en ambos casos con una edad promedio de 19 años.

Clasificación de palabras

Resuelve el siguiente ejercicio

Clasifica las palabras según correspondan

1



Fruta

2



Balón

3



Brócoli

Aguda 2

Grave 1

Esdrújula 3

Da clic en la celda de la que necesitas ayuda

Respuesta Correcta!



Ejercicio [1 de 3]
[Siguiente Ejercicio](#)

Figura 5.1. Ejercicio tipo de clasificación de palabras con base en su sílaba tónica en el tutor de Comunicación Oral y Escrita (Elaboración propia, proveniente del ambiente RITADA)

5.2 Instrumentos utilizados en el estudio de experiencia de usuario

Para la obtención de los datos se elaboró un cuestionario, adaptado del instrumento de usabilidad y experiencia de usuario UMUX-Lite y validado por un experto en psicología pedagógica. Cabe mencionar que el cuestionario se divide en tres áreas, la primera evalúa aspectos cognitivos en el estudiante, la segunda y tercera evalúan la aceptación y la eficiencia de la tutoría con base en el estilo de aprendizaje y la autoestima del estudiante, respectivamente. Específicamente, el instrumento generado está compuesto por 8 ítems de los cuales 2 son enfocados a obtener una retroalimentación general sobre el ejercicio realizado, 4 se centran en la apreciación de la asesoría considerando diversas categorías de estilos de aprendizaje, y los últimos 2 en la experiencia de usuarios con base en aspectos de autoestima. A continuación, se mencionan las preguntas del cuestionario utilizado:

- 1) ¿El ejercicio te resultó difícil de resolver?
- 2) ¿La retroalimentación proporcionada por el sistema te fue de utilidad?
- 3) ¿Las imágenes utilizadas para ejemplificar las preguntas y la retroalimentación fueron de utilidad?
- 4) ¿Consideras que es necesario incluir imágenes para ejemplificar las preguntas y la retroalimentación?
- 5) ¿Los ejemplos de respuesta correcta proporcionados en la retroalimentación fueron de gran utilidad?
- 6) ¿Consideras que es necesario incluir ejemplos de respuesta correcta en la retroalimentación?
- 7) ¿Cuándo una respuesta fue CORRECTA, qué mensaje te agradecería recibir?
- 8) ¿Cuándo una respuesta fue INCORRECTA, qué mensaje te agradecería recibir?

En el instrumento se utilizó una escala Likert con cinco respuestas para las preguntas 1 a 6, con las opciones: a) Totalmente de acuerdo, b) De acuerdo, c) Ligeramente de acuerdo, d) En desacuerdo, y e) Totalmente en desacuerdo. Para las preguntas 7 y 8 se utilizaron tres respuestas enfocadas en el nivel de autoestima. Para la pregunta 7: a) Respuesta correcta, b) Me da igual, y c) Excelente trabajo. Para la pregunta 8: a) Respuesta incorrecta, b) Me da igual, y c) Modifica tu respuesta.

El instrumento empleado para la recolección de los datos se automatizó en la plataforma RITADA para hacer más eficiente el acopio y análisis de la información recibida. Adicionalmente, en esta etapa se requirió utilizar nuevamente los instrumentos de diagnóstico, debido a que se propuso ofrecer una tutoría personalizada con base en el tipo de estilo de aprendizaje del alumno, así como en su nivel de autoestima. Tal como se describe con más detalle en la siguiente sección, los estudiantes contestaron los instrumentos de diagnóstico requeridos y posteriormente utilizaron el tutor inteligente del dominio de Comunicación Oral y Escrita, alojado en la plataforma RITADA. Finalmente, contestaron el instrumento de experiencia de usuario.

5.3 Procedimiento utilizado en el estudio de experiencia de usuario

El estudio final se realizó durante un mes, en el cual se estuvieron utilizando los módulos de diagnóstico y el módulo de COE. Se procedió a dividir el estudio final en dos fases, la primera fase sirvió para realizar el diagnóstico y la segunda fase para realizar una corrida de la arquitectura de software propuesta y llevar a cabo asesoría personalizada a los estudiantes, ofreciendo retroalimentación enfocada en su estilo de aprendizaje y el nivel de autoestima detectado previamente en la fase 1 del estudio.

El tiempo requerido para la ejecución de ambas fases fue de cuatro semanas, en las que los estudiantes estuvieron accediendo a la plataforma para registrarse, llevar a cabo el diagnóstico y utilizar la herramienta. En la parte del diagnóstico se integró la funcionalidad de un nuevo instrumento que permite obtener el estilo de aprendizaje enfocado en el modelo VAK (visual, auditivo, kinestésico) de Bandler y Ginder [70], o cualquiera de sus combinaciones. Aun cuando existen diferentes instrumentos y teorías para definir el estilo de aprendizaje de un estudiante, se empleó el instrumento VAK que se enfoca en los tres principales receptores sensoriales: visual, auditivo y kinestésico, el cual fue fácilmente adaptable a la plataforma digital de aprendizaje RITADA.

Al mismo tiempo, esta sección de RITADA ya contaba con el instrumento para obtener el nivel de autoestima de un estudiante. El instrumento utilizado en RITADA para medir la autoestima de una persona es el Inventario de Autoestima de Coopersmith [71], enfocado a obtener el nivel de autoestima bajo las categorías Familiar, Escolar y General. Los resultados originales generados por el instrumento implementado en RITADA fueron adaptados para generar la terna (FEG), clasificando cada categoría con un nivel de autoestima alto, medio o bajo (H, M y L, respectivamente). Considerando esencial la categoría escolar para personalizar la asesoría en la herramienta propuesta.

Durante la fase de diagnóstico se realizaron las siguientes actividades:

- 1) Capacitación a estudiantes en el uso y registro de la plataforma educativa RITADA.
- 2) Acceso a la plataforma RITADA.
- 3) Evaluación diagnóstica para la obtención de su nivel de autoestima y tipo de estilo de aprendizaje.
- 4) Ejecución y monitoreo constante del experimento.
- 5) Obtención de resultados y alimentación de la base de datos genérica del modelo del estudiante.

Para llevar a cabo la segunda fase de este estudio, una vez realizado el registro y el diagnóstico de los estudiantes, se dividieron los grupos por días para participar y hacer uso de la plataforma RITADA con la arquitectura de software implementada. Se procedió a implementar el algoritmo para dar seguimiento al proceso de asignación de asesoría de la plataforma RITADA. Particularmente, para la personalización de la asesoría brindada, el sistema obtiene datos provenientes del diagnóstico de la capa conductual. Específicamente se refiere a la terna que determina su estilo de aprendizaje y su nivel de autoestima. El algoritmo 1 describe el proceso de asignación de asesoría personalizada a cada estudiante una vez contestados los instrumentos de diagnóstico.

Algoritmo 1. Algoritmo utilizado para impartición de asesoría personalizada

1. El estudiante realiza el proceso de autenticación e ingresa al sistema.
2. El estudiante accede al módulo de ejercicios disponibles del tutor inteligente deseado.
3. El tutor inteligente determina el tipo de asesoría personalizada a brindar al estudiante.
 - 3.1. El tutor inteligente recupera la información del modelo del estudiante, disponible en la *base de datos compartida*. En este caso las tems VAK y FEG diagnosticadas, correspondientes al nivel de estilo de aprendizaje y autoestima.

- 3.2. El tutor inteligente consulta la base de conocimiento disponible en la ontología para determinar el tipo de asesoría a brindar al estudiante, considerando ambos criterios: estilo de aprendizaje y autoestima.
4. El estudiante selecciona y accede a un ejercicio de su interés de COE.
5. El tutor inteligente muestra un ejercicio personalizado de COE.
 - 5.1. El tutor inteligente consulta el nivel cognitivo del estudiante en dicho ejercicio de COE, disponible en la *base de datos propia* de ese dominio; en caso de existir actividad previa o vacío en caso de ser un ejercicio de COE nuevo para el estudiante.
 - 5.2. El tutor inteligente asigna un ejercicio de COE seleccionado de forma aleatoria, con base al nivel cognitivo del estudiante (nivel de complejidad básica, intermedia o alta).
 - 5.3. El tutor inteligente determina la forma de presentación del ejercicio de COE con base en las características personales del estudiante.
 - 5.3.1 Considerando el estilo de aprendizaje, el tutor inteligente determina si en la descripción del ejercicio de COE se utilizan imágenes, solo texto, o la combinación de ambos.
6. El estudiante realiza acciones para resolver el ejercicio de COE asignado.
7. El tutor inteligente monitorea la actividad del estudiante y determina proporcionar asesoría personalizada.
 - 7.1 Considerando el estilo de aprendizaje, el tutor inteligente determina si en la asesoría brindada se utilizan imágenes, solo texto, ejemplos tipo, o la combinación de ellos.
 - 7.2 Considerando el nivel de autoestima, el tutor inteligente determina el tipo de mensaje para indicar si una respuesta fue correcta o incorrecta.
8. El tutor inteligente almacena la actividad del estudiante en la base de datos propia: respuestas correctas e incorrectas, intentos, solicitud de ayuda, entre otros.
9. El tutor inteligente envía datos particulares al LRS, información a compartir con la comunidad de tutores inteligentes incluidos en la arquitectura propuesta.

Como se indica en el algoritmo, el tutor inteligente accede a la capa de conocimiento donde se encuentran los ejercicios de COE de la plataforma RITADA para asignar un ejercicio, la retroalimentación que ofrece el sistema es con base en el estilo de aprendizaje y nivel de autoestima. El estilo de aprendizaje determina el uso de imágenes, solo texto y

texto e imágenes (ver Figuras 5.2 y 5.3). Por otra parte, considerando la autoestima se cambia la forma de mostrar un error. Por ejemplo, el mensaje de *respuesta incorrecta* se intercambia por el mensaje de *modifica tu respuesta*, buscando que el mensaje sea más sutil al momento de que un estudiante con baja autoestima reciba la retroalimentación (ver Figura 5.4).

Clasificación de palabras

Resuelve el siguiente ejercicio

Clasifica las palabras según correspondan

1	2	3
		
Espectáculo	Veloz	Esfera
Aguda <input checked="" type="text" value="1"/>	Grave	Esdrújula

Ejercicio [1 de 3]
[Siguiente Ejercicio](#)

Da clic en la celda de la que necesitas ayuda

Respuesta Incorrecta...

Agudas: Llevan tilde al terminar con **N, S,** o **VOCAL**

Café
(última)



Figura 5.2. Ejercicio tipo de clasificación de palabras con base en su sílaba tónica utilizando texto e imágenes, en el tutor de Comunicación Oral y Escrita (Elaboración propia, proveniente del ambiente RITADA).

Resuelve el siguiente ejercicio

Clasifica las palabras según correspondan

1	2	3
Máquina	polar	comida
Aguda <input checked="" type="text" value="2"/>	Grave <input checked="" type="text" value="3"/>	Esdrújula <input checked="" type="text" value="1"/>

Ejercicio [1 de 3]
[Siguiente Ejercicio](#)

Da clic en la celda de la que necesitas ayuda

Excelente Trabajo!

Figura 5.3. Ejercicio tipo de clasificación de palabras con base en su sílaba tónica utilizando sólo texto, en el tutor de Comunicación Oral y Escrita (Elaboración propia, proveniente del ambiente RITADA).

Clasificación de palabras

Resuelve el siguiente ejercicio

Clasifica las palabras según correspondan

1 Poesía 2 Patinó 3 Ágiles

Aguda Grave Esdrújula

Da clic en la celda de la que necesitas ayuda
Modifica tu respuesta...

Ejercicio [1 de 3]
[Siguiente Ejercicio](#)

Figura 5.4. Ejercicio tipo de clasificación de palabras con base en su sílaba tónica utilizando mensaje sutil “modifica tu respuesta” sin imágenes, en el tutor de Comunicación Oral y Escrita (Elaboración propia, proveniente del ambiente RITADA).

En esta fase se considera de gran peso del uso de la ontología implementada, ya que es parte medular para la toma de decisiones de los tutores inteligentes del ambiente integral desarrollado. Al emplear el razonador de nuestra ontología, este decide qué tipo de ejercicios mostrará al alumno y que tipo de retroalimentación le dará. La ontología recibe la combinación proveniente del diagnóstico. Esta combinación es analizada por el razonador de la ontología, que al analizarla define qué tipo de combinación es y manda un valor numérico del 0 al 3 al sistema RITADA, indicando que tipo de retroalimentación y ejercicios ofrecerá al estudiante (ver figura 5.5), por otra parte en la tabla 5.1 se describen los valores que son enviados por la ontología y su representación.

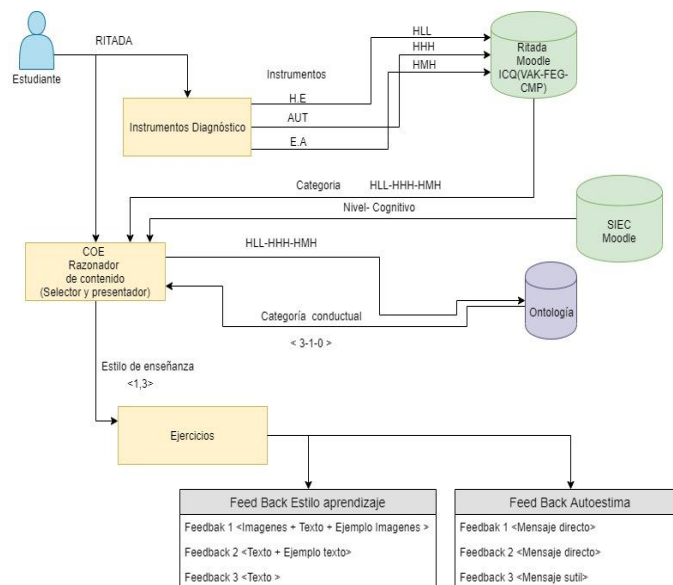


Figura 5.5. Asesoría personalizada por la arquitectura de software RITADA. (Elaboración propia).

5.4 Evaluación de aceptación de la arquitectura de software con base en el estilo de aprendizaje del estudiante

En esta sección se muestra la apreciación de los estudiantes de los 4 grupos participantes considerando las 2 preguntas generales y 4 dirigidas a evaluar la aceptación de la asesoría con base en estilo de aprendizaje (preguntas 3 a la 6). En las gráficas se observa que los estudiantes de cada grupo se presentan a través de 6 subgrupos, derivados de las 3 categorías puras de estilo de aprendizaje (visual, auditivo y kinestésico) y las 3 combinaciones resultantes de las mismas (visual-auditivo, visual-kinestésico y auditivo-kinestésico). En la Tabla 5.2 se presenta la distribución de estudiantes que fueron clasificados en cada subgrupo con base en sus respuestas al instrumento VAK. Como se puede observar en la Tabla 5.2, la mayor cantidad de estudiantes fueron clasificados dentro de las categorías puras, y un porcentaje muy bajo (0 a 7 estudiantes) son considerados dentro de las categorías de estilos de aprendizaje híbridas. Con base en lo anterior, el análisis de los resultados se realiza principalmente con las categorías puras a partir de la pregunta 3 del instrumento.

Tabla 5.1. Retroalimentación-Estilo de aprendizaje-Autoestima (elaboración propia).

Estilo de aprendizaje	#	Retroalimentación
Visual	1	Imágenes + Texto + Ejemplo en imagen
Auditivo	2	Texto + Ejemplo en texto
Kinestésico	3	Texto
Visual-Auditivo	3	Texto
Visual-Kinestésico	3	Texto
Auditivo-Kinestésico	3	Texto
Nivel de Autoestima		Retroalimentación
Alta	1	Mensaje directo
Media	2	Mensaje directo
Baja	3	Mensaje sutil

Tabla 5.2. Clasificación de estudiantes participantes en el estudio, con base en su estilo de aprendizaje predominante.

Estilo de Aprendizaje	Grupo 603	Grupo 605	Grupo 606	Grupo 731	Total
Visual	10	18	6	9	43
Auditivo	23	7	11	8	49
Kinestésico	1	2	3	5	11
Visual-Auditivo	1	3	2	1	7
Visual-Kinestésico	0	1	0	1	2
Auditivo-Kinestésico	0	1	0	1	2

En la Figura 5.6 se puede observar la opinión de los estudiantes sobre la primera pregunta del instrumento para evaluar la aceptación y satisfacción de la asesoría brindada, orientada a obtener retroalimentación general de un ejercicio: *¿El ejercicio te resultó difícil de resolver?* La mayoría de los estudiantes respondieron que el ejercicio no les resultó complicado de realizar. En 4 grupos se presenta que el 100% de los participantes indicaron que el ejercicio no fue difícil de realizar. La presencia de estudiantes que contestaron que los ejercicios son difíciles de realizar es casi nula, se presenta solamente en 4 grupos y en proporciones menores al 30%. Destacando que en 3 de estos grupos fueron los estudiantes con un estilo de aprendizaje auditivo los que aseveran esta apreciación, se asume que es derivado de la ausencia de instrucciones auditivas. Con relación a los estudiantes identificados con un estilo de aprendizaje visual puro, se observa que más del 70% consideraron que el ejercicio fue fácil de resolver, posiblemente por la predominancia de instrucciones visuales que permiten una mejor comprensión del ejercicio.

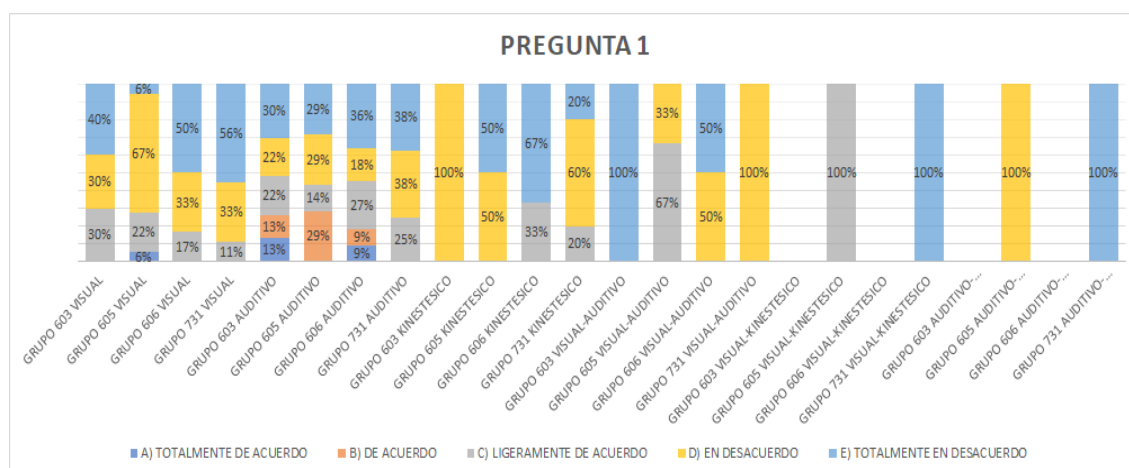


Figura 5.6. Pregunta 1 por grupo y estilo de aprendizaje. (Elaboración propia).

El mismo comportamiento se puede observar en la Figura 5.7, al responder la pregunta 2: *¿La retroalimentación proporcionada por el sistema te fue de utilidad?* Donde se aprecia que los estudiantes que participaron en la prueba contestaron principalmente estar totalmente de acuerdo y de acuerdo en que la retroalimentación recibida fue de gran utilidad. Se puede observar nuevamente que los estudiantes de 3 grupos, con estilo de aprendizaje auditivo, consideran en un porcentaje bajo, del 11%, 17% y 4%, que no fue de utilidad la retroalimentación (totalmente en desacuerdo y en desacuerdo). En esta ocasión sobresale un grupo de la categoría kinestésico con un 33% en desacuerdo en que la retroalimentación proporcionada fue de utilidad, y dos grupos bajo esta categoría indicando estar de acuerdo y ligeramente de acuerdo. De nuevo se considera que esta aseveración deriva por la ausencia de apoyo auditivo, y en el caso de los estudiantes bajo la categoría kinestésico por ser un ejercicio con prevalencia teórica.

Con relación a la pregunta 3: *¿Las imágenes utilizadas para ejemplificar las preguntas y la retroalimentación fueron de utilidad?*, en los grupos analizados dentro de la categoría visual, los únicos que recibieron instrucción y asesoría con imágenes, predominan las respuestas indicando que el uso de imágenes es de gran ayuda. Se observa en tres de estos grupos entre un 89% y 94% de los estudiantes visuales aceptando la utilidad de complementar texto con imágenes. Sin embargo, en el grupo 606 destaca un porcentaje significativo de estudiantes que no están totalmente convencidos de este beneficio (33% ligeramente de acuerdo), esto nos indica que el ambiente propuesto requiere considerar más variables para personalizar esta característica funcional del sistema (ver Figura 5.8).

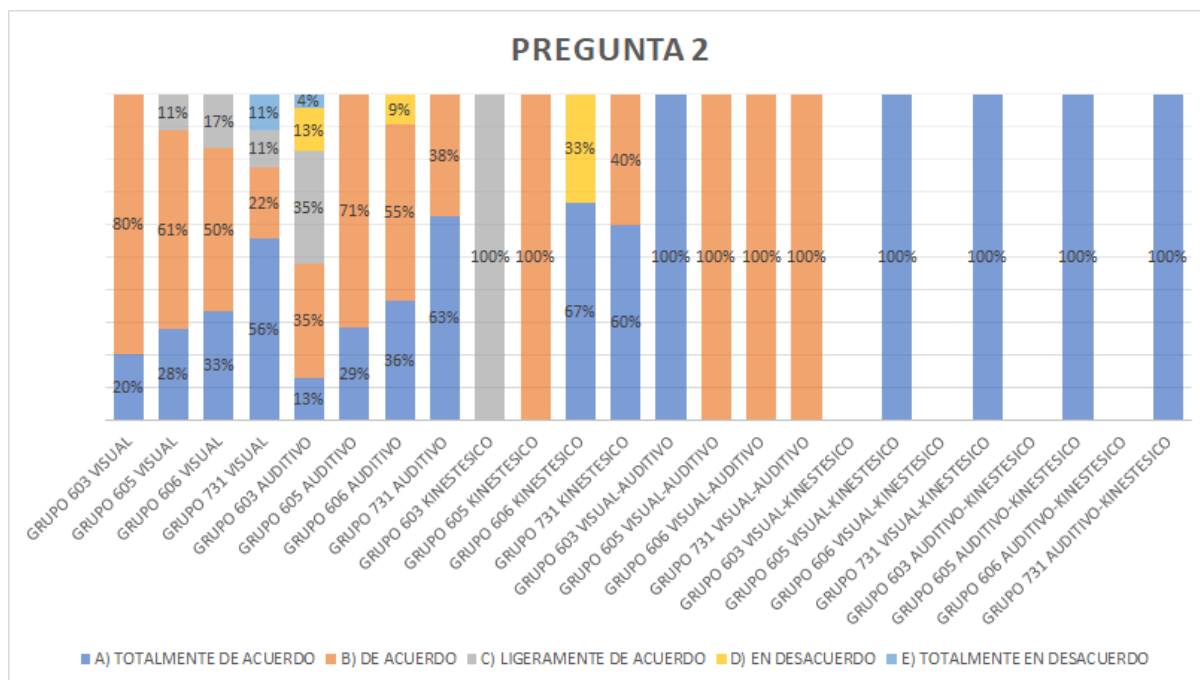


Figura 5.7. Pregunta 2 por grupo y estilo de aprendizaje. (Elaboración propia).

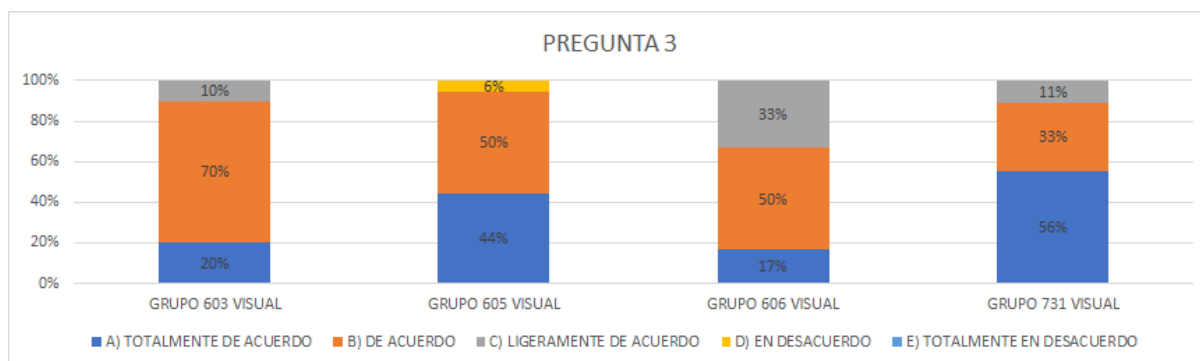


Figura 5.8. Pregunta 3 por grupo y estilo de aprendizaje. (Elaboración propia).

Para la pregunta 4: *¿Consideras que es necesario incluir imágenes para ejemplificar las preguntas y la retroalimentación?*, predominan los alumnos que indican como necesaria la retroalimentación de manera visual ya que les sirve de guía el observar los ejemplos para lograr un mejor desempeño. Sin embargo, analizando los resultados obtenidos por categoría de estilo de aprendizaje, podemos observar en tres grupos con alumnos de la categoría visual que alrededor del 10% de los estudiantes están ligeramente de acuerdo en el uso de imágenes y un porcentaje de hasta 30% de estudiantes que están en desacuerdo con su uso. Dentro de la categoría auditivo, tal como se esperaba, se reporta un porcentaje menor de estudiantes recomendando el uso de imágenes. Sin embargo, este

porcentaje se encuentra entre el 50% y 75%, siendo un resultado significativo considerando que este grupo de estudiantes fue clasificado en una categoría visual pura. Esto indica que, aun cuando el apoyo visual pudiera no ser crítico para este grupo de estudiantes, el uso de imágenes es aceptado y solicitado por ellos; posiblemente también influye la ausencia de instrucciones en audio.

Respecto a la categoría kinestésico, omitiendo el grupo 603 por contar con solo un estudiante, observamos que un porcentaje mayor de estos estudiantes no recomiendan el uso de imágenes. Esta respuesta pudiera atender al comportamiento práctico de los estudiantes kinestésicos, que en ocasiones proceden a realizar los ejercicios omitiendo la lectura de instrucciones o apoyos visuales; utilizándolos sólo cuando se enfrentan a un problema que les impide continuar con su actividad. Finalmente, es importante resaltar la respuesta de los estudiantes que fueron categorizados con el nivel de estilo de aprendizaje híbrido *visual-auditivo*, en este caso los alumnos reportan estar neutrales ante el uso de imágenes (ligeramente de acuerdo). Se asume que lo anterior deriva de su preferencia compartida ante un estilo u otro. Estos resultados, nuevamente reflejan que el estilo de aprendizaje debe ser considerado, pero no es el único factor para personalizar la presentación la instrucción y la asesoría. Con estos resultados se determina que el ambiente de aprendizaje debe ofrecer la asesoría considerando el diagnóstico inicial, pero este debe permitir al estudiante ajustar su personalización (ver Figura 5.9).

Por otra parte, en la Figura 5.10 se pueden observar las respuestas al contestar la pregunta 5: *¿Los ejemplos de respuesta correcta proporcionados en la retroalimentación fueron de gran utilidad?* Es importante indicar el antecedente de que los estudiantes bajo la categoría kinestésico no obtuvieron ejemplos de respuesta correcta, considerando que en este estilo de aprendizaje el alumno tiende a omitir instrucciones.

En los resultados obtenidos se puede apreciar que los estudiantes de ambas categorías analizadas consideran que los ejemplos de retroalimentación recibidos durante la elaboración de los ejercicios fueron de gran utilidad. Se hace hincapié que los ejemplos de apoyo que recibieron los estudiantes auditivos no incluyen imágenes. Por otra parte, se observa un par de grupos con un pequeño porcentaje de alumnos (10% y 13%) en desacuerdo con esta pregunta, solo presente en dos grupos evaluados. Con lo anterior se asume que el uso de ejemplos de apoyo en la asesoría es relevante, ya sea utilizando imágenes o sin ellas.

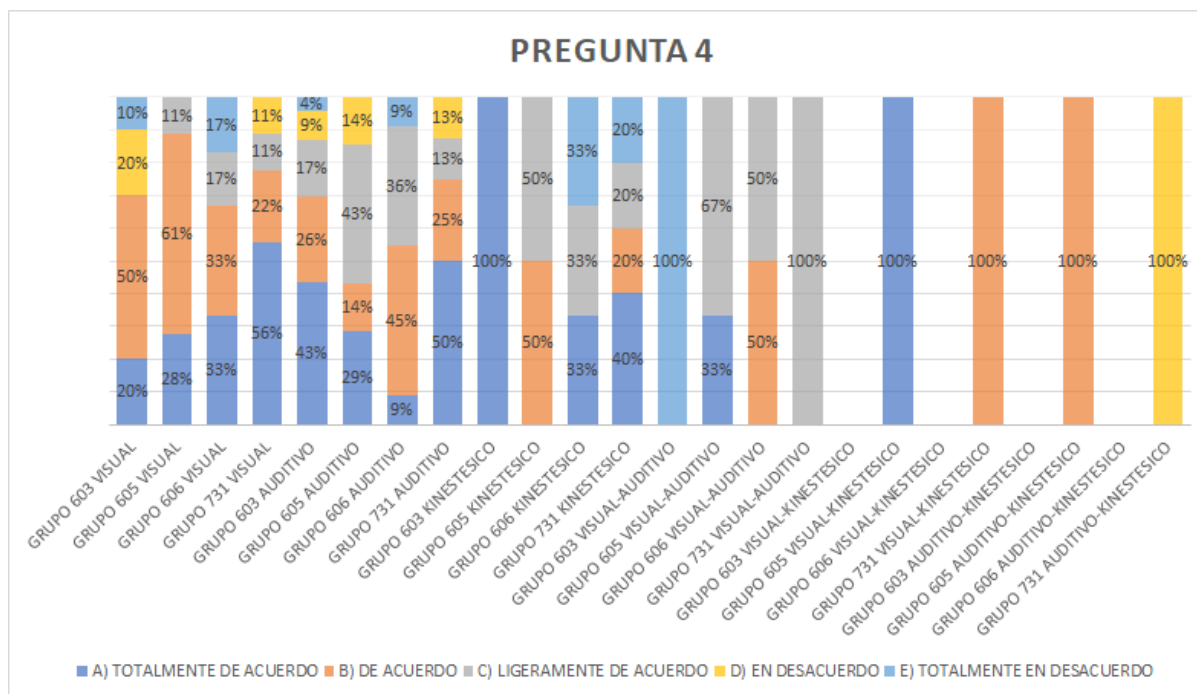


Figura 5.9. Pregunta 4 por grupo y estilo de aprendizaje. (Elaboración propia).

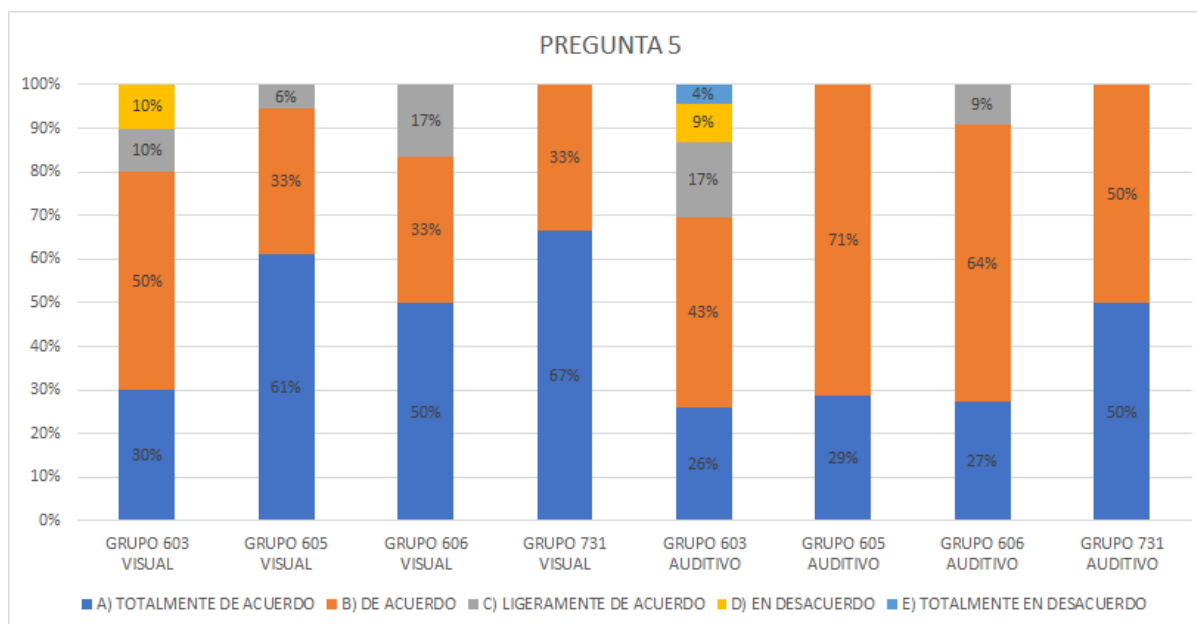


Figura 5.10. Pregunta 5 por grupo y estilo de aprendizaje. (Elaboración propia).

En la Figura 5.11 se puede observar el comportamiento de los grupos y su estilo de aprendizaje al contestar la pregunta 6: *¿Consideras que es necesario incluir ejemplos*

de respuesta correcta en la retroalimentación? Se puede observar que la mayoría de los estudiantes consideran necesario el que se incluyan ejemplos al recibir retroalimentación. El grupo con mayor porcentaje de alumnos que consideran innecesario el tener ejemplos de respuestas correctas lo encontramos en la clasificación kinestésico con un 20%, de nuevo asumiendo que a esta categoría de estudiantes prefieren explorar antes de recibir apoyo puntual.

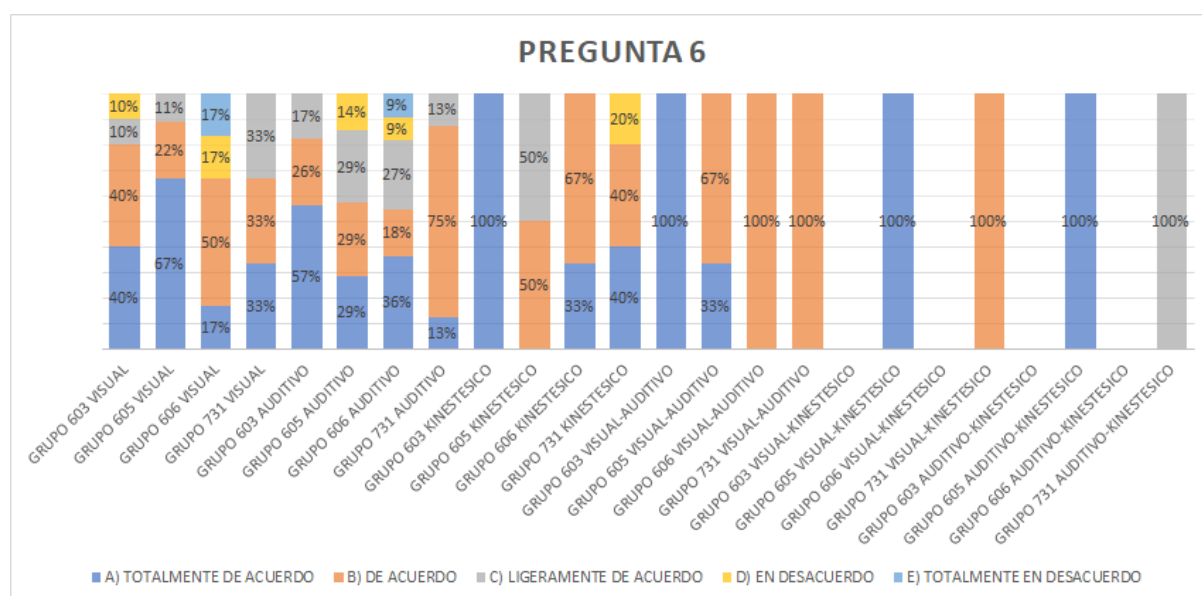


Figura 5.11. Pregunta 6 por grupo y estilo de aprendizaje. (Elaboración propia).

5.5 Evaluación de aceptación de la arquitectura de software con base en su nivel de autoestima

En esta sección se muestra la apreciación de los estudiantes de los 4 grupos participantes considerando las 2 preguntas generales y las 2 dirigidas a evaluar la aceptación de la asesoría con base en nivel de autoestima (preguntas 7 y 8). Similar a la sección anterior, en las gráficas se observa que los estudiantes de cada grupo se presentan a través de 2 subgrupos, derivados de los niveles de autoestima detectados en los estudiantes participantes (autoestima alta y media). En la Tabla 5.3 se presenta la distribución de estudiantes que fueron clasificados en cada subgrupo con base en sus respuestas al instrumento FEG. Como se puede observar en la Tabla 5.3, la mayor cantidad de estudiantes fueron clasificados equitativamente dentro de las categorías de

autoestima alta y media, solo se identificó un estudiante considerado dentro de la categoría de autoestima baja. Con base en lo anterior, el análisis de los resultados se realiza principalmente con las categorías con mayor incidencia a partir de la pregunta 3 del instrumento.

Tabla 5.3. Clasificación de estudiantes participantes en el estudio, con base en su nivel de autoestima.

Nivel de Autoestima	Grupo 603	Grupo 605	Grupo 606	Grupo 731	Total
Alta	17	27	18	12	74
Media	32	10	11	13	66
Baja	0	0	1	0	1

En la evaluación por grupo y por nivel de autoestima en la Figura 5.12 se puede observar que al responder la pregunta 1: *¿El ejercicio te resultó difícil de resolver?*, consistentemente con lo reportado en el apartado anterior, que considera el estilo de aprendizaje, independientemente del nivel de autoestima la mayoría de los estudiantes respondieron que el ejercicio no les resultó complicado de realizar. La presencia de estudiantes que contestaron que el ejercicio fue difícil de realizar es casi nula, se presenta solamente en proporciones menores al 18%.

El mismo comportamiento se presenta para la pregunta 2: *¿La retroalimentación proporcionada por el sistema te fue de utilidad?*, se aprecia en la gráfica que la mayor parte de los estudiantes que participaron en la prueba contestaron que la retroalimentación recibida fue de gran utilidad. Sin embargo, se observan dos grupos con un porcentaje menor de aceptación de la utilidad de la asesoría, los grupos corresponden a estudiantes con nivel de autoestima media; 603 y 606 con 60% y 63%, respectivamente (ver figura 5.13).

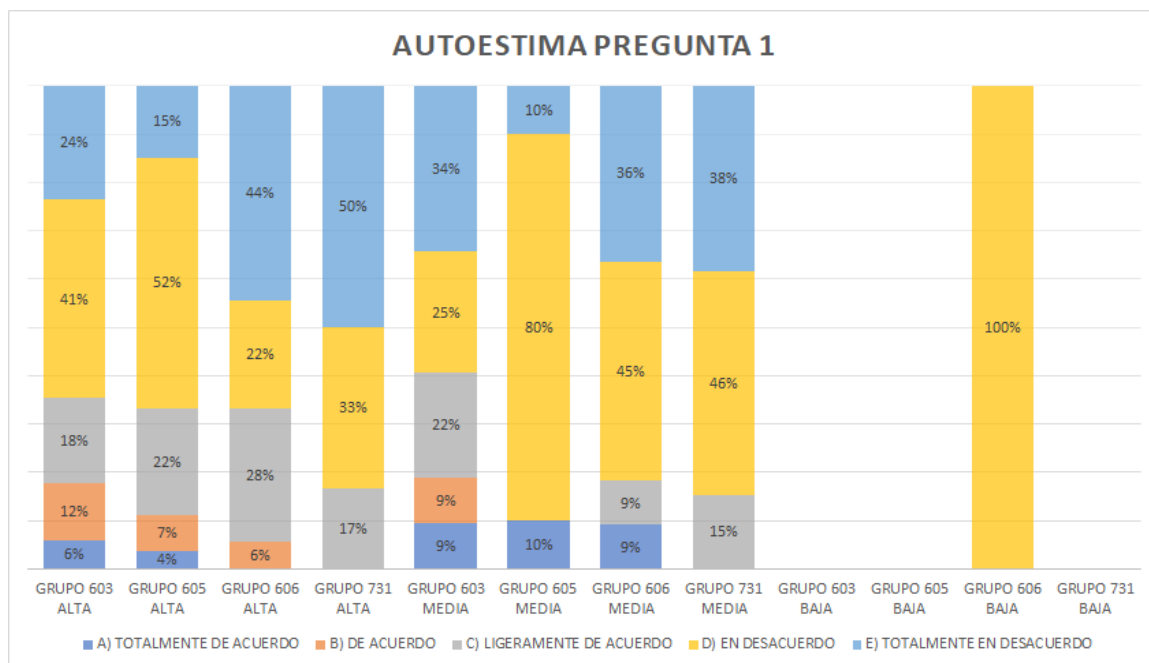


Figura 5.12. Pregunta 1 por grupo y nivel de autoestima. (Elaboración propia).

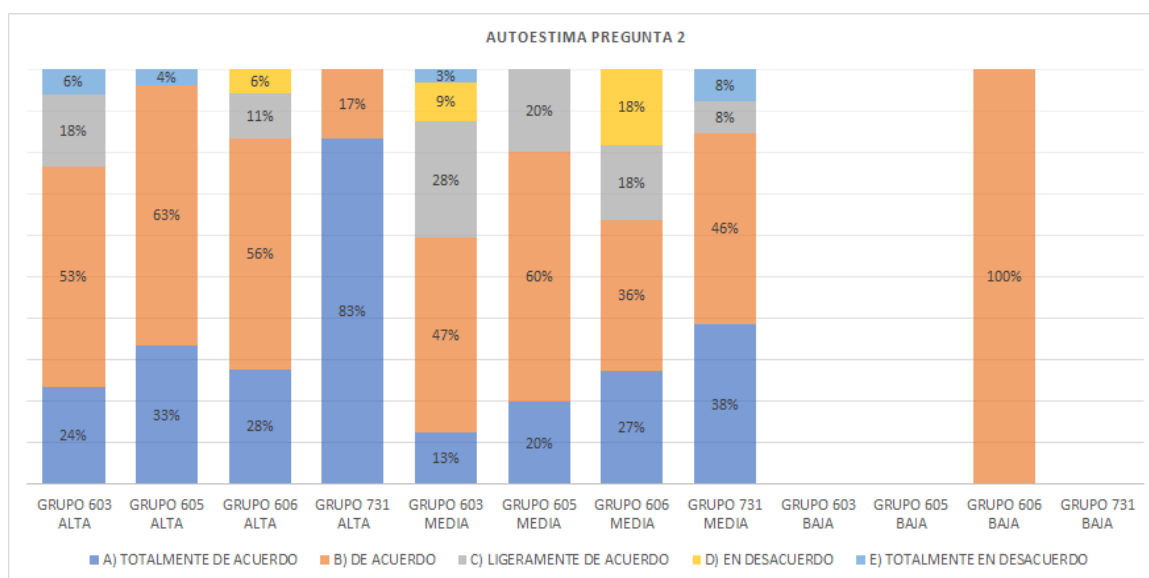


Figura 5.13. Pregunta 2 por grupo y nivel de autoestima. (Elaboración propia).

En la figura 5.14 se ve el comportamiento al contestar la pregunta 7 orientada a medir el tono de la retroalimentación: *¿Cuándo una respuesta fue CORRECTA qué mensaje te agradaría recibir?* Los resultados indican que los alumnos se sienten más cómodos al recibir un mensaje estimulante reconociendo su trabajo como *Excelente Trabajo*, y se observa un ligero incremento en los grupos de estudiantes con autoestima media. Un porcentaje de estudiantes parecido indicó que les gusta el reconocimiento con mensajes directos, como el de *Respuesta Correcta*. Cabe mencionar que se obtuvo un porcentaje muy pequeño que reportan indiferencia con el tono del mensaje de reconocimiento de logros obtenidos.

Aun cuando sólo se identificó un estudiante con autoestima baja, es importante enfatizar que su respuesta dio preferencia al uso de mensaje motivante ante una respuesta correcta. Con base en estos resultados, se determina que es importante definir un tono de mensaje de reconocimiento de logros, sin embargo, el nivel de autoestima obtenido al momento no resultó ser un elemento contundente que ayude a determinarlo.

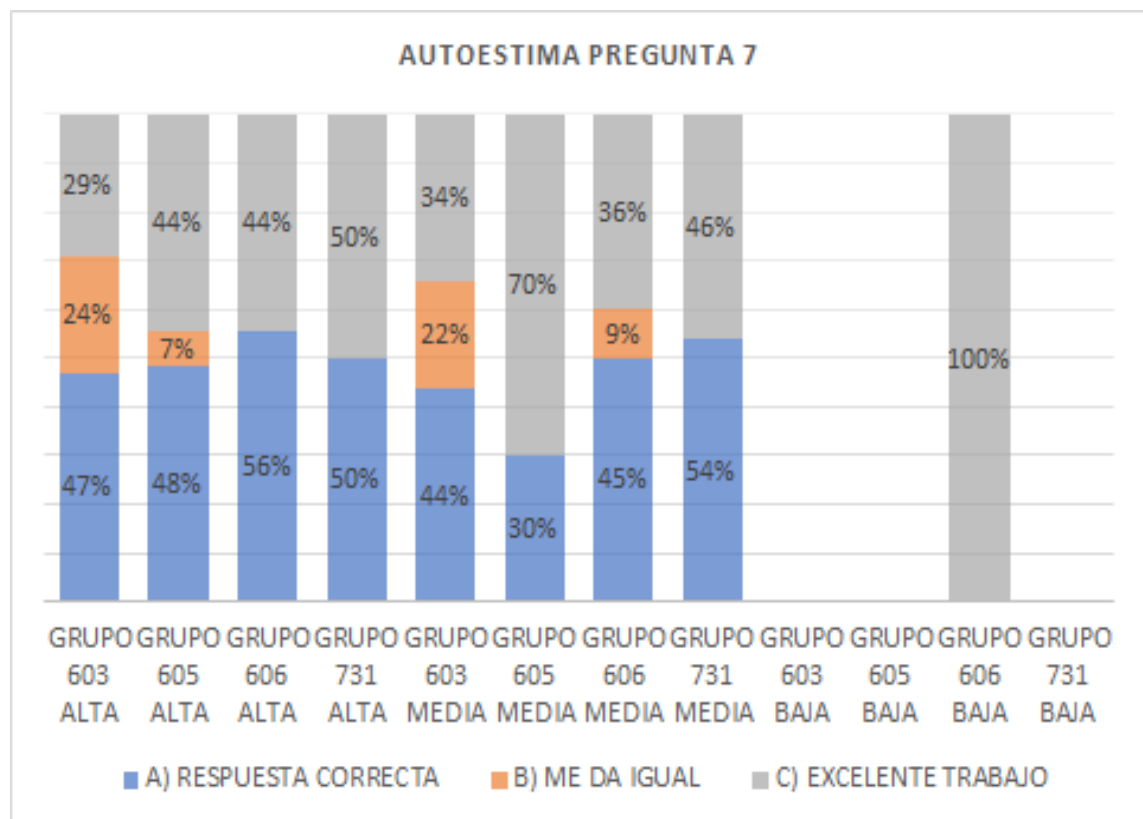


Figura 5.14. Pregunta 7 por grupo y nivel de autoestima. (Elaboración propia).

En la figura 5.15 se puede apreciar la tendencia al contestar la pregunta 8: *¿Cuándo una respuesta fue INCORRECTA qué mensaje te agradaría recibir?* De los grupos de estudiantes evaluados se logra observar que aun cuando la mayoría de los estudiantes indican sentirse más cómodos con un mensaje más sutil al realizar una acción incorrecta, son los estudiantes con una autoestima media los que en mayor porcentaje aprueban este tono de asesoría. Con respecto al estudiante identificado con autoestima baja, su respuesta indica que prefiere un mensaje sutil cuando comete un error. En este caso en particular, se recomienda usar un tono sutil al indicar un error en estudiantes con autoestima media o baja, y opcional para los estudiantes para autoestima alta. Para concluir esta etapa, se determina que se puede trabajar la autoestima y los estilos de aprendizaje de forma independiente para brindar asesoría y se pueden combinar para personalizar la asesoría brindada.

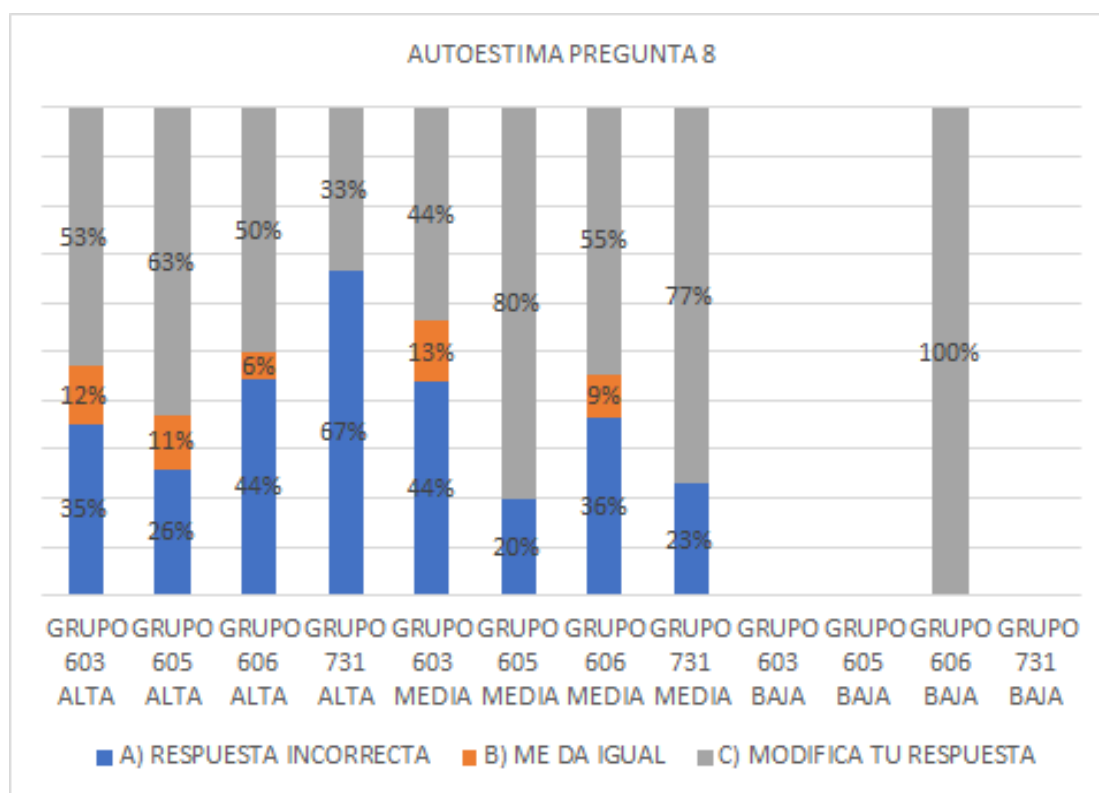


Figura 5.15. Pregunta 8 por grupo y nivel de autoestima. (Elaboración propia).

Capítulo 6

Conclusiones y trabajo futuro

Esta investigación describe el diseño, implementación y evaluación de una arquitectura de software que permite la comunicación entre tutores inteligentes, permitiendo homogeneizar el modelo del estudiante para ser compartido entre tutores inteligentes de diferentes dominios. La arquitectura desarrollada es distinta a ambientes de tutoría inteligente encontrados en la literatura, ya que las investigaciones consultadas se enfocan en el desarrollo de los módulos de los tutores inteligentes de forma aislada, ofreciendo el servicio de asesoría de manera personalizada pero independiente. En el presente capítulo se describen las conclusiones derivadas de la presente investigación, aportaciones al área de investigación y algunas sugerencias de trabajo futuro.

6.1 Conclusiones:

Con base en la investigación realizada, podemos concluir que el desarrollar una arquitectura de software que permita la comunicación entre tutores inteligentes, así como homogeneizar el modelo del estudiante es un trabajo no explorado, sin embargo, los estudios encontrados sirvieron de base para guiar el desarrollo de esta investigación. A continuación, se presentan las conclusiones derivadas de este trabajo organizadas por objetivos específicos y general.

Para atender el primer objetivo específico, *“establecer un esquema de representación de conocimiento para la implementación del modelo del estudiante que permita su integración con modelos de experto independientes del dominio”*, se puede aseverar que se logró el resultado esperado al modelar el conocimiento del estudiante facilitando su uso en múltiples tutores inteligentes. Se implementó el modelo para representar el conocimiento a través de una ontología, así como los algoritmos para consumir el conocimiento almacenado en dicho modelo.

En la presente investigación, el desarrollo de la ontología se consideró de suma importancia considerando que es el componente que nutre de información al tutor

inteligente al momento de tomar decisiones para asesorar a estudiantes de forma personalizada. Se tomó la decisión de utilizar modelos ontológicos debido a que la base de conocimiento a representar es considerada un problema complejo, ya que el conocimiento y los factores que determinan el comportamiento del estudiante al aprender es un problema multifactorial; requiere ser validado por expertos humanos y al mismo tiempo permitir el entendimiento de los conceptos por la computadora.

Como resultado de la revisión de la literatura, aun cuando se encuentra un amplio trabajo en el modelado ontológico en diversos dominios, se observó que existe poca información referente al uso de ontologías en el área de tutores inteligentes. En la fase inicial de la presente investigación se logró contactar a una investigadora del Ecuador que ha realizado trabajos de modelado ontológico para tutores inteligentes. Adicionalmente, se recopilaron algunos trabajos de investigación orientados a la educación usando este tipo de modelado. Debido a los requerimientos de nuestra investigación no se consideró pertinente utilizar ninguna ontología de las disponibles en la literatura, ya que estaban diseñadas para atender aspectos muy específicos de dominios particulares. Sin embargo, fueron consideradas como base en el diseño de la ontología propuesta.

Con base en las bondades y resultados ofrecidos por los modelos ontológicos revisados para otras áreas de estudio, se llegó a la determinación de que la representación del conocimiento mediante una ontología es factible y permite establecer un modelo para compartir conocimiento entre ambientes heterogéneos. Al mismo tiempo, el uso de esta técnica de modelado permitió obtener una representación escalable, es decir, poder incrementar o ajustar la base de conocimiento atendiendo las necesidades particulares de uno o varios ambientes educativos, de forma transparente para el resto y sin afectar o requerir realizar ajustes en sus funcionalidades.

También, se concluyó que las características genéricas del estudiante, tales como su estilo de aprendizaje, autoestima, así como técnicas de enseñanza, se pueden modelar a través de la web semántica. Lo anterior permitió que el uso del modelo ontológico propuesto pueda ser utilizado en ambiente web. Finalmente, se enfatiza que, aun cuando la representación del conocimiento y aspectos conductuales del estudiante se considera un problema complejo, el software Protege permitió la representación y creación de la ontología requerida. Al utilizar el software Protege fue posible representar el modelo del estudiante y, mediante la implementación de algoritmos específicos, se logró consultar la

ontología e inferir en las decisiones que toma el tutor inteligente al dar asesoría personalizada.

En el caso del segundo objetivo específico, *“crear una arquitectura de software que permita la cooperación e interacción entre múltiples tutores inteligentes, al compartir su modelo del estudiante”*, se determinó el desarrollar una arquitectura integral que permitiera la comunicación e interacción entre tutores inteligentes heterogéneos. Al hacer la revisión de la literatura no se encontró una arquitectura como tal en el ámbito de los sistemas de tutoría inteligente.

Un paso fundamental para desarrollar la arquitectura propuesta fue el considerar el uso de la ontología implementada y una máquina de inferencia basada en reglas para permitir la comunicación entre los tutores heterogéneos. Basándonos en la literatura sobre agentes inteligentes, se generó una arquitectura multiagente deliberativa en la que los agentes participantes en la arquitectura tienen acceso a la información generada por los otros agentes, poniendo la información que cada agente genera disponible para ser consumida por el resto. Al mismo tiempo, se implementó una interacción implícita abierta, es decir cooperación bajo demanda sin solicitud, lo cual nos llevó a generar un ecosistema de aprendizaje orientado a contenidos y bases de conocimiento provenientes de diversas fuentes de información.

La arquitectura de software propuesta consiste de tres capas principales. La primera capa incluye el conjunto de sistemas de tutoría inteligente o módulos educativos creados para el usuario final, desarrollados en forma independiente y heterogénea, sin ser previamente diseñados para interactuar con otros sistemas. La segunda sección corresponde a la capa de los modelos o representaciones de conocimiento. En el caso particular de la arquitectura implementada se encuentran los modelos del dominio propios de cada tutor (conocimiento del experto) y la ontología que representa el modelo compartido del estudiante y pedagógico. En la tercera capa se encuentran los bancos de datos de los estudiantes, orientados al conocimiento que gestiona de forma particular cada tutor inteligente, así como el banco de datos compartido que almacena información genérica del estudiante. Con base en el trabajo realizado, se concluye que la arquitectura colaborativa propuesta requiere únicamente el establecer un banco de datos centralizado donde se concentre la base de conocimiento del estudiante que contiene características

genéricas (aspectos conductuales y emocionales), así como aspectos pedagógicos donde se especifique el tipo y la forma de proporcionar la retroalimentación.

Por otra parte, para el tercer objetivo específico, *“establecer un protocolo de comunicación entre tutores inteligentes, basado en la tecnología multiagentes”*, previo al establecimiento de un protocolo propietario, se revisó la literatura y se encuentra con la reciente propuesta del protocolo Experience API, cuyo objetivo principal es comunicar sistemas heterogéneos para compartir servicios y/o recursos educativos. Se exploró sobre las características del estándar xAPI y se consideró adecuado para cubrir los objetivos de comunicación e interacción de la arquitectura de tutores inteligentes propuesta. Se tomó la decisión de utilizar xAPI por las bondades que ofrece la especificación, siendo una de sus principales premisas que el aprendizaje se da en cualquier parte y busca que toda experiencia de aprendizaje sea registrada utilizando un formato estándar, esto permite que la información generada sea reutilizada al interactuar sistemas heterogéneos.

Sin embargo, considerando que a la fecha es un tema poco explorado, se enfrentó el problema de poca información y documentación al respecto. Al realizar la revisión de blogs de especialistas en el tema, se encontró y contacto a un investigador en el área proveniente de la Ciudad de México, el cual nos orientó en aspectos técnicos para utilizar xAPI y en la toma de algunas decisiones al seleccionar librerías y definir mecanismos para implementar aplicaciones que permitieron la comunicación entre tutores inteligentes.

Al utilizar la especificación xAPI se logró entablar una comunicación estable entre tutores inteligentes a través del almacén de conocimiento LRS (Learning Locker), al mismo tiempo permitió la comunicación entre las diferentes plataformas (tutores inteligentes) sin importar el lenguaje en el que estuvieran programadas. Se concluye que el protocolo de comunicación xAPI permite una adecuada interacción entre tutores inteligentes heterogéneos, cumpliendo de manera satisfactoria las expectativas de comunicación al ser integrado en la arquitectura de software propuesta manteniendo estable la comunicación sin importar el número y procedencia de los usuarios.

Al atender el objetivo específico cuatro, *“lograr un desempeño operativo de la arquitectura propuesta al interactuar con múltiples tutores inteligentes de diversos dominios, considerando sus características de operatividad, rendimiento, compatibilidad, y escalabilidad”*, se procedió a integrar y evaluar la operación de todos

los elementos seleccionados y desarrollados para la arquitectura propuesta: tutores inteligentes previamente desarrollados, protocolo de comunicación xAPI, almacén de conocimiento LRS, modelo del estudiante compartido, bancos de datos particulares y genéricos. La evaluación abarcó los aspectos de interés para considerar una funcionalidad aceptable de la arquitectura: operatividad, rendimiento, compatibilidad y escalabilidad.

En la evaluación de funcionalidad de la arquitectura se contó con la participación de más de 400 estudiantes provenientes de 11 grupos de nivel superior y medio superior. Los detalles de los resultados del estudio son presentados en el capítulo cuatro de este documento. En el estudio se encontró que para la prueba de rendimiento se obtuvo un acceso concurrente máximo de 106 peticiones simultáneas, donde el sistema soportó las peticiones sin interrumpir sus servicios y sin degradar su tiempo de respuesta. Se observó que el protocolo de comunicación utilizado, especificación xAPI y el repositorio LRS, permiten que la interacción entre tutores y cada uno de sus componentes se lleve a cabo de manera correcta y transparente, siendo compatible con los componentes que integran el sistema actualmente (tutores inteligentes).

Durante la prueba de operatividad se consideró la capacidad de compartir información entre el modelo del dominio y el modelo del estudiante a través del protocolo utilizado por el LRS y la capacidad de uso de la información entre múltiples dominios. A este respecto obtuvimos un resultado favorable ya que el intercambio de información se da de forma adecuada, no existe pérdida de información y la información obtenida se encuentra siempre disponible para ser consumida. Por otra parte, el sistema se consideró compatible y escalable, es decir puede crecer y replicarse independientemente del ambiente de desarrollo de las aplicaciones que se integren. Con lo anterior se concluye que, mediante el uso de la especificación xAPI, el LRS y el conocimiento modelado a través de una ontología, la plataforma RITADA funcionando mediante la arquitectura propuesta, y los tutores inteligentes que la componen, operan de manera estable.

Al atender el objetivo específico cinco, “*evaluar la aceptación y eficiencia de la tutoría brindada a estudiantes mediante la arquitectura de tutores inteligentes propuesta*”, similar a las actividades descritas en el objetivo anterior, se procedió a trabajar con todos los componentes de la arquitectura de software de manera integral. En esta etapa de la investigación, para lograr el objetivo cinco, fue necesario medir el nivel de aceptación y la eficiencia de la tutoría brindada según la apreciación de usuarios reales.

Para llevar a cabo la evaluación fue indispensable la interacción de los participantes con la plataforma RITADA. En esta ocasión se contó con la participación de 4 grupos de la Facultad de Ciencias Administrativas y Sociales de la UABC, en los que participaron un total de 141 alumnos. Los estudiantes fueron previamente capacitados en el uso de la plataforma, posteriormente se procedió a evaluar su percepción ante la asesoría brindada por el tutor inteligente que cuenta con ejercicios y brinda apoyo en el área de comunicación oral y escrita. Los resultados de la evaluación realizada a los grupos de estudio son presentados en el capítulo 5 de este documento. En el estudio se encontró que el personalizar la asesoría considerando aspectos conductuales permite a los alumnos obtener retroalimentación acorde a sus necesidades, permitiéndonos el análisis al comparar las respuestas dadas en los grupos divididos por estilos de aprendizaje y nivel de autoestima.

Durante esta fase del estudio se consideraron las 3 áreas definidas en el instrumento para evaluar la aceptación y la eficiencia de la asesoría con base en su estilo de aprendizaje y nivel de autoestima. Se concluye que ambos factores se pueden trabajar de forma independiente para brindar asesoría, y se puede generar combinaciones para generar una mejor personalización de la tutoría brindada. Al medir el nivel de aceptación de la asesoría que proporcionó el tutor inteligente, se concluye que existe un comportamiento muy similar en la respuesta de los estudiantes independiente de los estilos de aprendizaje y los niveles de autoestima identificados. Finalmente, los resultados obtenidos al considerar la autoestima al proporcionar asesoría, se detectó que los alumnos se sienten más cómodos con mensajes motivadores cuando realizan una acción correcta y mensajes sutiles para referirse a errores cometidos al intentar solucionar los ejercicios presentados en la plataforma RITADA, además los estudiantes se sienten cómodos y consideran de gran ayuda la asesoría personalizada recibida.

Al considerar el objetivo general *“desarrollar y evaluar la funcionalidad de una arquitectura basada en software, robusta y escalable, que soporte un ambiente interactivo de tutores inteligentes, permitiendo la cooperación y coordinación de las actividades de asesoría, así como homogeneizar y compartir la base de conocimiento del estudiante”*, se concluye que fue posible el desarrollo de una arquitectura con las características propuestas. La arquitectura resultante integra la ontología que modela el conocimiento de estudiantes y aspectos pedagógicos, el LRS Learning Locker que

permite el almacenar experiencias de usuario basado en un mismo lenguaje, los tutores inteligentes multidominio basados en desarrollos heterogéneos y el uso de LMS como Moodle. A su vez, la arquitectura es robusta en su operación y escalable al permitir la integración de nuevos tutores inteligentes sin afectar su funcionamiento. En su entorno operativo, se evidenció que el ambiente se puede utilizar de manera simultánea por los participantes y la arquitectura se mantiene estable. Por otra parte, la cooperación entre tutores inteligentes es posible debido a que se compartió el modelo del estudiante entre los tutores participantes en la arquitectura, al mismo tiempo que cada tutor coordina las actividades propias de asesoría para sus estudiantes.

6.2 Áreas de oportunidad y trabajo futuro

La asesoría personalizada se considera de suma importancia en la tutoría tradicional y virtual. Debido a las limitantes que existen en el modelo tradicional de tutorías es complicado llegar a un gran número de estudiantes sin afectar la calidad de la tutoría, con este trabajo se propone una base para la construcción de una arquitectura de software que permita la integración de múltiples tutores inteligentes que compartan su conocimiento particular de cada estudiante en beneficio de una mejor personalización, y en consecuencia de la calidad al impartir asesoría.

Una vez concluido el estudio, establecido hace cuatro años, y considerando los resultados favorables obtenidos, el objetivo actual es extender esta investigación atendiendo las siguientes áreas de oportunidad:

- Realizar un estudio del impacto académico ante la asesoría personalizada brindada por la plataforma propuesta, comparado con los resultados ofrecidos por la versión previa de la misma plataforma y contrastándola con la asesoría convencional (profesor-alumno).
- Implementar un módulo de analítica de aprendizaje para obtener los resultados filtrados y en forma gráfica del desempeño funcional del ambiente y del desempeño académico de los estudiantes, categorizados por las diferentes combinaciones entre estilo de aprendizaje, autoestima y hábitos de estudio.

- Integrar más aspectos conductuales y emocionales al modelo del estudiante representado en la ontología resultante de la presente investigación.
- Promover la integración de más tutores inteligentes a la arquitectura.

Referencias

- [1] Belloch, C. (2012). Las Tecnologías de la Información y Comunicación en el aprendizaje. Departamento de Métodos de Investigación y Diagnóstico en Educación. Universidad de Valencia.
- [2] Tello, E. (2007). Las tecnologías de la información y comunicaciones (TIC) y la brecha digital: su impacto en la sociedad de México. *International Journal of Educational Technology in Higher Education (ETHE)*, **4(2)**.
- [3] Koedinger, K.R. y Corbett, A.T. (2006). Cognitive tutors: Technology bringing learning sciences to the classroom. In: *The Cambridge handbook of the learning sciences*, R.K. Sawyer, Editor. Cambridge University Press: New York, NY, **61-77**.
- [4] Álvarez-Xochihua, O., Muñoz-Merino, P.J., García-Pericuesta, R., Kloos, C.D., & Gutiérrez-Lopez, E. (2017). Learning Analytics Implementation in a Multidomain Computer-Based Learning Environment. congreso LASI, España
- [5] ANUIES. (2001). Deserción, rezago y eficiencia terminal en las IES: propuestas metodológica para su estudio. Asociación Nacional de Universidades e Instituciones de Educación Superior.
http://books.google.com.mx/books/about/Desert%C3%B3n_rezago_y_eficiencia_terminal.html?hl=es&id=9jl0AQAACAAJ (consultado el 06 de Abril de 2018).
- [6] Zapata et al (2018)
http://reaxion.utleon.edu.mx/Art_Examen_de_seleccion_Vs_desercion_escolar_en_nivel_superior_Una_aproximacion_a_la_realidad_universitaria.html (Consultado en 07 de noviembre de 2019)
- [7]Secretaria de educación pública (2019) consultado noviembre de 2019 en https://www.planeacion.sep.gob.mx/Doc/estadistica_e_indicadores/principales_cifras/principales_cifras_2017_2018_bolsillo.pdf.
- [8] Gómez U, J. M. (2000). Factores que influyen en el alto índice de reprobación en las materias de ciencias básicas de la Facultad de Ingeniería Mecánica y Eléctrica (Tesis Doctoral, Universidad Autónoma de Nuevo León).
- [9] Contreras, F., Espinosa, J. C., Esguerra, G., Haikal, A., & Polanía, A. (2005). Autoeficacia, ansiedad y rendimiento académico en adolescentes. *Diversas: Perspectivas en Psicología*, **183-194**.

- [10] Flores-Payán S., Camacho-Ríos A. , (2014). Análisis estadístico de las causas de reprobación desde la perspectiva del alumno de Ingeniería del Instituto Tecnológico de Chihuahua II.
- [11]Chávez, C. R. (2018). El compromiso del docente desde la percepción de los estudiantes de licenciatura en administración de una IES.
- [12] Romo, A (2003). La incorporación de los programas de tutoría en la Instituciones de Educación Superior. México:ANUIES
- [13]Suárez(2019)consultado el 20 de septiembre de 2016 <https://www.elsoldemexico.com.mx/mexico/sociedad/reforma-universidades-anuies-reforma-educativa-3102133.html>
- [14]D`Ambrosio (1993) consultado el 16 de abril 2016 <https://dialnet.unirioja.es/servlet/articulo?codigo=6436481>
- [15]Lozada, R. G. (2001). Factores que inciden en la aplicación de estrategias docentes para el aprendizaje significativo del alumno de Educación Básica. Telos: Revista de Estudios Interdisciplinarios en Ciencias Sociales, 3(2), 193-210.
- [16]Cruz, S(2017) Consultado el 13 de agosto 2019, <http://fcays.ens.uabc.mx/wp-content/uploads/2019/02/181218-Informe-Fcays-2017-2018.pdf>
- [17] Woolf, B. P., & Cunningham, P. (1987). Building a community memory for intelligent tutoring systems (pp. 82-89).
- [18] Cataldi, Z. (2009). Sistemas Tutores Inteligentes Orientados a la Enseñanza para la Comprensión. EDUTEC Revista Electrónica de Tecnología Educativa (p.5)
- [19] Vanlehn, K (1988). Student Modelling. M. Polson. Foundations of Intelligent Tutoring systems. Hillsdale. N.J. Lawrence Erlbaum Associates, (pp. 55-78)
- [20] Ferrer, A., Vallejos, F. A., Jiménez Rey, E. M., Grossi, M. D., Servetto, A. C., & Perichinsky, G. (2006). Modelo de sistemas tutores inteligentes multiagente. In VIII Workshop de Investigadores en Ciencias de la Computación.
- [21] Carbonell, J. R. (1970). AI in CAI: An artificial intelligence approach to computer assisted instruction. IEEE transaction on Man Machine System. Vol.11, Nro. 4, p. 190-202.
- [22] Wolf, B. (1984). Context Dependent Planning in a Machine Tutor. Ph.D. Dissertation, University of Massachusetts, Amherst, Massachusetts.
- [23] Brown, S. & Burton, R. R, de Kleer. (1982) Pedagogical, natural language and knowledge engineering techniques in Sophie In D. Sleeman and J. S. Brown, editors, ITS, 227-282, N.Y, Ac. Press.

- [24] Matich, D. J. (2001). Redes Neuronales: Conceptos básicos y aplicaciones. Cátedra de Informática Aplicada a la Ingeniería de Procesos–Orientación I.
- [25] Matich, D. J. (2001). Redes Neuronales: Conceptos básicos y aplicaciones. Cátedra de Informática Aplicada a la Ingeniería de Procesos–Orientación I.
- [26] Cataldi, Z., Salgueiro, F., & Lage, F. (2006). Predicción del rendimiento de los estudiantes y diagnóstico usando redes neuronales. enseñanza,(pp. 14, 15)
- [27] Pane, J. F., Griffin, B. A., McCaffrey, D. F., & Karam, R. (2014). Effectiveness of cognitive tutor algebra I at scale. Educational Evaluation and Policy Analysis, **36(2)**, 127-144.
- [28] Tavares, D. B. L., Xochihua, O. Á., Jordan-Aramburo, A., & Sánchez-Guzmán, D. (2016). Sistema de Tutores Inteligentes para Ayudar a Estudiantes de Bachillerato en la Solución de Problemas de Matemáticas. Latin-American Journal of Physics Education, **10(4)**, 12.
- [29] Barrón Estrada, M. L., Zatarain Cabada, R., & Hernández Pérez, Y. (2014). Tutor Inteligente con reconocimiento y manejo de emociones para Matemáticas. Revista electrónica de investigación educativa, **16(3)**, 88-102.
- [30] Puga, J. L., & García, J. G. (2008). Sistemas de tutorización inteligente basados en redes bayesianas. REMA Revista electrónica de metodología aplicada, **13(1)**, 13-25.
- [31] Petrovica, S., Anohina-Naumeca, A., & Ekenel, H. K. (2017). Emotion Recognition in Affective Tutoring Systems: Collection of Ground-truth Data. Procedia Computer Science, **104**, 437-444.
- [32] Tello, A. L. (2001). Ontologías en la Web semántica. España: Universidad De Extremadura.
- [33] Flórez-Fernández, H., 2007. Construcción de ontologías OWL. Vínculos. Ciencia Tecnología y Sociedad: un enlace hacia el futuro, **vol. 4, no. 1, pp. 19-34**.
- [34] Paul Buitelaar y Philipp Cimiano, “Ontology Learning and Population: Bridging the Gap between Text and Knowledge”, IOS Press. Nieuwe Hemweg 6B, 1013 BG Amsterdam, Netherlands, **vol. 167, 2008**.
- [35] Gruber, T. (1993). What is an Ontology. WWW Site <http://www-ksl.stanford.edu/kst/whatis-an-ontology.html>
- [36] Abián M. 2005. “El Futuro de la Web, XML, RDF/RDFS, ontologías y la Web Semántica”. Dublin Core Metadata Element Set (DCMES), Versión 1.1 (2012). Consultado en Julio 2017 en <http://www.dublincore.org/documents/dces/>

- [37]Rodríguez, C. M., Montaña, W. C., & Martínez, J. M. (2010). Razonadores semánticos: un estado del arte. Ingenium, (21).
- [38]Gascueña, J. M., Fernández-Caballero, A., & González, P. (2005). Ontologías del modelo del alumno y del modelo del dominio en sistemas de aprendizaje adaptativos y colaborativos. Laboratory of User Interaction and Software Engineering (LoUISE), Instituto de Investigación en Informática de Albacete, Universidad de Castilla-La Mancha.
- [39]Paneva, D. (2006). Use of Ontology-based Student model in Semantic-oriented Access to the Knowledge in Digital Libraries. In proc. of HUBUSKA Fourth Open Workshop “Semantic Web and Knowledge Technologies Applications”, Varna, Bulgaria (pp. 31-41).
- [40]Panagiotopoulos, I., Kalou, A., Pierrakeas, C., & Kameas, A. (2012, September). An ontology-based model for student representation in intelligent tutoring systems for distance learning. In IFIP International Conference on Artificial Intelligence Applications and Innovations, Springer Berlin Heidelberg (pp. 296-305).
- [41]Stuart J. Russell and Peter Norvig. (2003). Artificial Intelligence: A Modern Approach. Prentice Hall, 2 edition.
- [42]Jacques Ferber (1999). Multi-agents System and introduction to distributed Artificial. Addison-Wesley.
- [43]J. E. Beck, B. P. Woolf y C. R. Beal, ADVISOR; A machine learning architecture for improving collaboration: the DEGREE approach. International Journal of Artificial Intelligence in Education, págs.**221 –241,2000**
- [44]Maes P. (1989). Situated Agents Can Have Goals, “Designing Autonomous Agents: Theory and Practice From Biology to”.
- [45]Cámara, de la Fuente Lidia (2004). La representación lingüística del conocimiento y su relevancia en la ingeniería lingüística. "Hipertext.net", **núm. 2, 2004**.
- [46]Keith D., Pannu A., Sycara K. y Williamson M. (1997). Designing Behaviors for Information Agents“, Autonomous Agent
- [47]Rizo, R., Llorens, F., Pujol, M., & Artificial, I. (2003). Arquitecturas y comunicación entre Agentes. Departamento de Ciencia de la Computación e Inteligencia Artificial, Universidad de Alicante.
- [48]Wooldridge, M. (2009). An introduction to multiagent systems. John Wiley & Sons.
- [49] Llorens, F. (2014). Campus virtuales: de gestores de contenidos a gestores de metodologías. Revista de Educación a Distancia, **(42),1-12**.

- [50] Uden, L., Wangsa, I. & Damiani, E. (2007). The future of B-learning: B learning ecosystem. En Inaugural IEEE International Conference on Digital Ecosystems and Technologies. Cairns: IEEE (pp. 113-117).
- [51] García, F.J., Hernández, Á., Conde, M.Á., Fidalgo, Á. & Sein, M.L. (2015). Mirando hacia el futuro: Ecosistemas tecnológicos de aprendizaje basados en servicios. Congreso Internacional sobre aprendizaje, innovación y competitividad. Madrid: Fundación General de la Universidad Politécnica de Madrid. (pp. 553-558).
- [52] García-Holgado, A. & García-Peñalvo, F. (2017). Definición de ecosistemas de aprendizaje independientes de plataforma. IV Congreso Internacional sobre Aprendizaje, Innovación y Competitividad. Zaragoza.
- [53] Montenegro, C.E., Gaona, P.A., Cueva, J.M. & Sanjuán, Ó. (2011). Application of model-driven engineering (mda) for the construction of a tool for domain-specific modeling (dsm) and the creation of modules in learning management systems (lms) platform independent. *Dyna*, **78** (169), 43-52.
- [54] Hruska, M., Medford, A. & Murphy, J. (2015). Learning Ecosystems Using the Generalized Intelligent Framework for Tutoring (gift) and the Experience API (xAPI). *aied 2015 Workshop Proceedings*, junio 22-26. Madrid.
- [55] Sottilare, R.A., Brawner, K.W., Goldberg, B.S., & Holden, H.K. (2012). The generalized intelligent framework for tutoring (gift). Orlando: US Army Research Laboratory-Human Research & Engineering Directorate (arl-hred).
- [56] Rustici Software. (2018). What is the Experience api? Recuperado de <https://experienceapi.com/overview/>
- [57] Evolución de Tin Can API (2015) consultado en Noviembre 2017 (<https://experienceapi.com/tin-can-evolution/>)
- [58] Borra y Esguerra, A. B. (2016) Game for Learning: An Application of Tin Can API and Learning. De La Salle University, Manila, Philippines.
- [59] Bower, B. L., & Hardy, K. P. (2004). From correspondence to cyberspace: Changes and challenges in distance education. *New Directions for Community Colleges*, 128.
- [60] Lozano, M. P. (2011). Introducción al modelo de Referencia SCORM. Zaragoza: e-colección
- [61] Versiones de Scorm (2017). Consultado en diciembre de 2017. <https://scorm.com/scorm-explained/business-of-scorm/scorm-versions/>
- [62] Qué es Experience API (2017) consultado en Septiembre 2017 <https://experienceapi.com/overview/>

- [63] Evolución de Tin Can API (2015). <https://experienceapi.com/tin-can-evolution/> consultado en Noviembre 2017
- [64] Especificación Experience Api (2013) http://www.adlnet.gov/wp-content/uploads/2013/10/xAPI_v1.0.1-2013-10-01.pdf. consultado en Septiembre 2017
- [65] Qué es un LRS (2011) <https://scorm.com/what-is-an-lrs-learning-record-store/> consultado Septiembre 2017
- [66] Learning record Store consultado Septiembre 2017 <https://experienceapi.com/learning-record-store/>
- [67] Hernández Sampieri, R., Fernández Collado, C., & Baptista Lucio, P. (2014). Metodología de la investigación: Roberto Hernández Sampieri. Carlos Fernández Collado y Pilar.
- [68] Lewis, J. R., Utesch, B. S., & Maher, D. E. (2013, April). UMUX-LITE: when there's no time for the SUS. In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (pp. 2099-2102). ACM.
- [69] Damián-Reyes, P., Gutiérrez-Pulido J.R., Acosta-Díaz, R., y Herrera Morales, J.R.(2018). Sistema de tutoría inteligente basado en la contextualización del usuario como una estrategia de apoyo a estudiantes en el nivel de educación superior. Universidad Autónoma de Baja California.
- [70] Jorge Neira Silva, “Visual, Auditivo y kinestésico los alumnos”, en Orientación Andújar, N°1,p5.<https://www.orientacionandujar.es/2015/09/02/test-de-estilos-de-aprendizaje-de-vak-escolar-infantil-primaria-y-secundaria/visual-auditivo-o-kinestésico-los-alumnos>.
- [71] Prewitt-Diaz, J. (1984). A Cross-Cultural Study of the Reliability of the Coopersmith Self Esteem Inventory. Educational and Psychological Measurement, (pp. 575-581)

Anexos

Anexo 1

Glosario de términos

Términos	Conceptos
ADL	Aprendizaje Avanzado Distribuido
AI	Inteligencia Artificial
ANUIES	Asociación de Universidades e Instituciones de Educación Superior
CMS	Sistema de gestión de cursos.
DL	razonador de Lógica Descriptiva
FCAyS	Facultad de Ciencias Administrativas y sociales
FEG	Familiar, Escolar, General
GIFT	Marco Inteligente Generalizado para Tutoría
H	Alta
ILS	Sistema integrado de aprendizaje.
ITS	Sistema de tutoría inteligente.
ITS	Sistema de Tutoría Inteligente
L	Baja
LMS	Sistema de Gestión de Aprendizaje
LMS	Sistema de Gestión de Aprendizaje
LP	Plataforma de aprendizaje.
LRS	Contenedor de Registros de Aprendizaje
LSS	Sistema soporte aprendizaje.
M	Media
MDA	Arquitectura Dirigida de Modelos
MLE	Ambiente controlado de aprendizaje.
RDF	Marco de descripción de recursos
RITADA	Red de Innovación Tecnológica en Ambientes Digitales

SES	Sistema de Educación Superior
SITI	Sistema Integral de Tutoría Inteligente
TIC	Tecnologías de la Información
UABC	Universidad Autónoma de Baja California
URI	Identificador de Recursos Uniformes
VAK	Visual, Auditivo, Kinestésico
VLE	Entornos virtuales de aprendizaje.