

**DISEÑO E IMPLEMENTACION DE UN SISTEMA TUTOR INTELIGENTE
BASADO EN WEB APLICADO A LA RESOLUCIÓN DE ECUACIONES
ALGEBRAICAS NO LINEALES UTILIZANDO MÉTODOS NUMÉRICOS PARA
INGENIERÍA.**

**LEONARDO JOSE REYES DIAGO
ANDRES DARIO DE LA OSSA ARROYO**

**UNIVERSIDAD TECNOLOGICA DE BOLIVAR
FACULTAD DE INGENIERIAS
PROGRAMA DE INGENIERIA DE SISTEMAS
CARTAGENA DE INDIAS, D. T. y C.**

2008

**DISEÑO E IMPLEMENTACION DE UN SISTEMA TUTOR INTELIGENTE
BASADO EN WEB APLICADO A LA RESOLUCIÓN DE ECUACIONES
ALGEBRAICAS NO LINEALES UTILIZANDO MÉTODOS NUMÉRICOS PARA
INGENIERÍA.**

**LEONARDO JOSE REYES DIAGO
ANDRES DARIO DE LA OSSA ARROYO**

**PROYECTO DE GRADO PARA OPTAR AL TITULO DE
INGENIERO DE SISTEMAS**

**MOISES R. QUINTANA ALVAREZ
Licenciado en Matemáticas
Msc. En Informática Aplicada
DIRECTOR**

**UNIVERSIDAD TECNOLOGICA DE BOLIVAR
FACULTAD DE INGENIERIAS
PROGRAMA DE INGENIERIA DE SISTEMAS
CARTAGENA DE INDIAS, D. T. y C.**

2008

NOTA DE ACEPTACIÓN

Firma Comité de Investigación

Firma del Evaluador

Firma del Evaluador

Cartagena de Indias D.T. y C., Bolívar, Colombia

A los _____ días de _____ de _____.

AUTORIZACION

Yo, LEONARDO JOSE REYES DIAGO identificado con cédula de ciudadanía no. 73.006.516, autorizo a la Universidad Tecnológica de Bolívar, hacer buen uso de mi trabajo de grado y hacerlo público en el catálogo Web de la biblioteca.

LEONARDO JOSE REYES DIAGO

AUTORIZACION

Yo, ANDRES DARIO DE LA OSSA ARROYO identificado con cédula de ciudadanía no. 1'104.008.083, autorizo a la Universidad Tecnológica de Bolívar, hacer buen uso de mi trabajo de grado y hacerlo público en el catálogo Web de la biblioteca.

ANDRES DARIO DE LA OSSA ARROYO

Cartagena de Indias, D. T. y C., 10 de Octubre de 2008

Señores:

COMITÉ DE INVESTIGACIONES

Universidad Tecnológica de Bolívar

Facultad de Ingenierías

Programa de Ingeniería de Sistemas

Ciudad

Cordial saludo.

Adjunto a la presente hacemos entrega del proyecto de grado titulado **DISEÑO E IMPLEMENTACION DE UN SISTEMA TUTOR INTELIGENTE BASADO EN WEB APLICADO A LA RESOLUCIÓN DE ECUACIONES ALGEBRAICAS NO LINEALES UTILIZANDO MÉTODOS NUMÉRICOS PARA INGENIERÍA** como requisito para optar al título profesional en Ingeniería de Sistemas.

Esperamos de su parte una respuesta satisfactoria y agradecemos su amable atención.

Lo anterior fue aprobado y revisado por el director de proyecto.

Atentamente,

LEONARDO J. REYES DIAGO
C. C. No. 73.006.516

ANDRES D. DE LA OSSA ARROYO
C. C. No. 1'104.008.083

Cartagena de Indias, D. T. y C., 10 de Octubre de 2008

Señores:

COMITÉ DE INVESTIGACIONES

Universidad Tecnológica de Bolívar

Facultad de Ingenierías

Programa de Ingeniería de Sistemas

Ciudad

Cordial saludo.

En calidad de director del proyecto de grado titulado **DISEÑO E IMPLEMENTACION DE UN SISTEMA TUTOR INTELIGENTE BASADO EN WEB APLICADO A LA RESOLUCIÓN DE ECUACIONES ALGEBRAICAS NO LINEALES UTILIZANDO MÉTODOS NUMÉRICOS PARA INGENIERÍA** elaborado por los señores Leonardo José Reyes Diago y Andrés Darío De la Ossa Arroyo, manifiesto que he participado en la orientación del desarrollo del mismo en todas sus etapas y por consiguiente estoy totalmente de acuerdo con los resultados obtenidos.

Atentamente,

MOISES R. QUINTANA ALVAREZ

Licenciado en Matemáticas

Msc. En Informática Aplicada

CONTENIDOS

INTRODUCCIÓN.....	1
1. PROBLEMA DE LA INVESTIGACIÓN	5
1.1. PLANTEAMIENTO DEL PROBLEMA	5
1.2. FORMULACIÓN DEL PROBLEMA: EL SISTEMA TUTOR INTELIGENTE COMO COMPLEMENTO A LA PRESENCIALIDAD	7
2. JUSTIFICACIÓN.....	9
2.1. OBJETO.....	13
3. OBJETIVOS.....	14
3.1. OBJETIVO GENERAL	14
3.2. OBJETIVOS ESPECÍFICOS	14
4. ALCANCE.....	16
5. MARCO TEÓRICO	18
5.1. INFORMÁTICA EDUCATIVA	18
5.2. INTELIGENCIA ARTIFICIAL	21
5.2.1. DEFINICIÓN	21
5.2.2. HISTORIA	22
5.2.3. APLICACIÓN EN LA EDUCACIÓN.....	28
5.3. SISTEMAS TUTORES INTELIGENTES	30
5.3.1. DEFINICIÓN	30

5.3.2. COMPONENTES DE UN STI	32
5.3.3. FUNCIONAMIENTO DE UN STI.....	41
5.3.4. LOS SISTEMAS HIPERMEDIA ADAPTATIVOS.....	43
5.4. MÉTODOS NUMÉRICOS PARA LA SOLUCIÓN DE ECUACIONES ALGEBRAICAS NO LINEALES.....	48
5.4.1. CONCEPTOS PRELIMINARES.....	48
5.4.2. DEFINICIÓN DE MÉTODO NUMÉRICO	58
5.4.3. ERRORES	59
5.4.4. CIFRAS SIGNIFICATIVAS.....	61
5.4.5. ESTABILIDAD Y CONVERGENCIA	62
5.4.6. TOLERANCIA.....	63
5.4.7. MÉTODOS CERRADOS.....	65
5.4.8. MÉTODOS ABIERTOS.....	72
6. MARCO METODOLÓGICO.....	86
7. DISEÑO DEL SISTEMA TUTOR INTELIGENTE.....	88
7.1. ESTRUCTURA DE OBTENCIÓN DEL CONOCIMIENTO	89
7.2. DISEÑO DE PREGUNTAS A FORMULAR.....	93
7.3. ARQUITECTURA DEL SISTEMA TUTOR.....	100
7.3.1. ASPECTOS DE DISEÑO.....	100
7.3.2. INTERFAZ GRÁFICA (GUI).....	102
7.3.3. SERVICIO WEB.....	103
7.3.4. TUTOR.....	104
7.3.5. MÁQUINA DE INFERENCIA.....	105

7.3.6. BASE DE DATOS DEL ALUMNO	106
7.3.7. BASE DE DATOS DE CONOCIMIENTO	107
7.4. DISEÑO DE ESTRUCTURAS DE DATOS	108
8. CONCLUSIONES	110
FUENTES DE REFERENCIA	115
A N E X O S	122

ÍNDICE DE TABLAS

Tabla 1. Módulos de un STI.....	12
Tabla 2. Criterios de adaptabilidad en un Sistema Hipermedia Adaptativo	45
Tabla 3. Métodos de adaptación.....	46
Tabla 4. Tabla de iteraciones del método de Bisección.....	70
Tabla 5. Tabla de iteraciones del método de Falsa Posición.....	71
Tabla 6. Tabla de iteraciones del método del punto fijo.....	81
Tabla 7. Tabla de iteraciones del método de Newton-Raphson.	83
Tabla 8. Tabla de iteraciones del método de la secante.....	84

ÍNDICE DE FIGURAS

Figura 1. Estados de aprendizaje en un STI.	12
Figura 2. Arquitectura de un Sistema Tutor Inteligente	32
Figura 3. Aplicaciones de los sistemas hipermedia adaptativos.	47
Figura 4. Límite por la derecha de una función.	49
Figura 5. Límite por la izquierda de una función.	50
Figura 6. Representación gráfica del teorema del valor medio.	56
Figura 7. Estructura de obtención de conocimiento.	90
Figura 8. Conjuntos de partida y llegada para una pregunta de tipo I.	96
Figura 9. Conjuntos de partida y llegada para una pregunta de tipo II.	97
Figura 10. Conjuntos de partida y llegada para una pregunta de tipo III.	98
Figura 11. Conjuntos de partida y llegada para una pregunta de tipo V.	99
Figura 12. Arquitectura del STI implementado en la investigación.	102
Figura 13. Selección de un nodo siguiente.	106

ÍNDICE DE IMÁGENES

Imagen 1. Gráfica de la función $f(x) = x^3 + 4x^2 - 10$	69
Imagen 2. Vecindad de la función $f(x) = x^3 + 4x^2 - 10$ en el intervalo $[1,2]$	80
Imagen 3. Ejemplo de pregunta de tipo 'complete el enunciado'	95
Imagen 4. Ejemplo de una pregunta de tipo 'Selección múltiple con múltiple respuesta'	99

INTRODUCCIÓN

La educación es un concepto tan antiguo como el hombre y así como él, es cambiante a lo largo de la historia. Está ligado de tal forma que se vale de cada tecnología que el hombre crea con el fin de mejorarla.

El esquema que se tiene de la educación es el de un hombre impartiendo enseñanza a una comunidad ansiosa de conocimiento en un espacio real, generalmente cerrado tal como un aposento al que se conoce como aula. Este proceso es cíclico pues un individuo perteneciente a esa comunidad en la cual se brindó enseñanza, puede estar en capacidad de transmitir ese conocimiento a otros y así trasciende a otros individuos que también podrán hacer lo mismo con él. No obstante, este saber no es una verdad absoluta, por lo tanto se requiere seguirlo estudiando para transformarlo o mantenerlo según lo demanden las circunstancias.

La tecnología entra en escena como instrumento educativo, un apoyo, una facilidad. Exactamente hablando, a las comunicaciones, herramientas de vital importancia para el conocimiento, puesto que es inconcebible que el hombre se mantenga fuera de su realidad. Ejemplos como la educación por correspondencia,

radio o televisión facilitan el aprendizaje y sirven a aquellas personas que están limitadas para trasladarse a un sitio de encuentro real.

En la actualidad, el concepto ha cambiado de forma más no de fondo. Un elemento más se ha incorporado al entorno educativo: la computadora.

Ésta ofrece una oportunidad como un recurso más para la instrucción. Pero, para el objetivo que se ilustró anteriormente, una sola computadora no es de gran ayuda si se quiere enseñar a una comunidad. Aunque se implementen herramientas de software educacionales, éstas actuarán a manera de libros de texto, sin contribuir al esquema alumno-profesor porque la enseñanza aquí es unipersonal. Es aquí donde las redes y más aún la Internet, le dan trascendencia a la computadora pues una serie de ellas que puedan comunicarse entre sí revive el esquema de educación que se presentó inicialmente; sin embargo, lo altera significativamente en un aspecto: ya no será imperiosa la reunión presencial para transmitir conocimiento. El concepto de aula como un lugar físico ha quedado desvirtuado.

Si se habla de la pedagogía como la ciencia que se ocupa de los problemas educativos, entonces la informática educativa es la rama de aquella que se ocupa de las aplicaciones educativas de las herramientas informáticas¹ como las

¹ Castellano M., Hugo – Informática Educativa – Versión Electrónica <En Línea> URL: <http://cie.ilce.edu.mx/sitio/academica/Inform%E1tica%20educativa.pdf> Pág. 1.

computadoras y las herramientas de software que manejan, haciendo alusión al tema propuesto.

Enlazando este último punto con la noción de aula transformada por la aparición de la computadora, es necesario explorar cómo se puede enseñar utilizando las herramientas computacionales. Esta indagación concluirá en el siguiente axioma: el concepto de profesor encarnado en una persona real también cambiará al igual que cambió el concepto de aula, porque la computadora también será capaz de enseñar.

Este postulado, descrito en el párrafo anterior, es la concepción de este documento. A lo largo del mismo se explorarán recursos para alcanzar tal meta. Pero, ¿qué herramientas enmarcadas dentro de la informática educativa podrán emplearse para ello?

Al ser tan vasto el campo de la informática pueden pensarse muchas alternativas para educar, como emplear el correo electrónico, posibilidad moderna a la educación por correspondencia sin las trabas que implica el no saber si el mensaje se envió o se recibió, o los múltiples foros que existen en la Internet para el intercambio de ideas. Sin embargo, lo que se busca es una manera más directa o más bien, más interactiva, tal como suele ocurrir en el esquema tradicional de la educación. Aparecen entonces, herramientas más complejas como los sistemas tutores inteligentes que permiten la revisión y evaluación constante del aprendizaje

según el nivel de abstracción que tengan maestro y alumno obviando el inconveniente de la ya mencionada reunión presencial.

Es importante precisar que, en ningún motivo, se quiere implementar esta posible solución como la respuesta final y definitiva para el problema de la educación, o decir que el profesor será reemplazado definitivamente –hablando en tiempo presente, ya que a futuro esta perspectiva podría cambiar- por una computadora, pues mientras se siga explorando en el campo de la inteligencia artificial, siempre será necesario –al menos por ahora- la existencia de una persona que siga estudiando y revisando el conocimiento a fin de demostrar si aún es vigente o necesario de mejoras.

Haciendo referencia al campo aplicativo de este documento, el objeto de esta investigación es indagar una manera de enseñar métodos numéricos asociados a la solución de ecuaciones algebraicas no lineales fundamentados en los principios del aula virtual y la interactividad educativa. La respuesta que se implementará, reiterando el párrafo anterior, será un sistema tutor inteligente, que además de lo expuesto permite el desarrollo incremental de la actividad cognoscitiva, es decir, el conocimiento se aprenderá paso a paso acompañado el alumno por su ‘tutor virtual’ de modo que se constate que aquel ha comprendido cabalmente lo que se le ha enseñado o, en caso de falencias en algún tema que ya se haya visto pueda retroceder para reforzar ese saber confuso, sin perder de vista lo que se le ha presentado.

1. PROBLEMA DE LA INVESTIGACIÓN

1.1. PLANTEAMIENTO DEL PROBLEMA

Los métodos numéricos son ampliamente utilizados en el ámbito de la ingeniería. El hecho que se presenten problemas de ciencias como la física o la química que no puedan ser resueltos de forma analítica, sino que requieran de un procedimiento numérico hace fehaciente tal premisa.

No obstante, su enseñanza en la Universidad Tecnológica de Bolívar, como caso objetivo, sólo está destinada al estudiante de Ingeniería de Sistemas, como rigurosidad del pénsum del programa. Para los demás programas de pregrado de Ingeniería, esta asignatura es electiva. Se requiere que el estudiante de Ingeniería que desee cursar esta materia, haya cumplido sus ciclo básico de matemáticas (Álgebra, Cálculo Integral, Cálculo Vectorial, Ecuaciones Diferenciales).

Su instrucción se imparte de manera presencial haciendo particular énfasis en el procedimiento iterativo (la algoritmia) para encontrar la solución de problemas. A causa de ello, la computadora aparece como agente colaborador, asistiendo principalmente, a aquellos estudiantes que presenten falencias en programación.

Hoy, se emplean aplicaciones de software que facilitan el cómputo y el análisis de los métodos numéricos. Sin embargo, son aplicaciones de 'caja negra',² en las cuales sólo es necesario ingresar lo que se debe calcular para obtener un resultado y no cómo se consiguió, contrario a la noción de aprendizaje que se desea cuando se está en un proceso de obtención de conocimiento. Aquí no se contribuye a la enseñanza ya que el alumno no es testigo del desarrollo secuencial del resultado del problema planteado.

En algunas ocasiones, como proyectos dentro de la metodología de la asignatura, se propone a los estudiantes escribir sus propios programas que reflejen los pasos necesarios para la consecución de soluciones a cualquier problema que aplique para el método numérico que se esté analizando.

Existen otras herramientas computacionales que posibilitan el flujo de enseñanza-aprendizaje. Una de ellas, los sistemas tutores inteligentes, mejor conocidos como STI, alcanzan este objetivo permitiendo la escalabilidad del conocimiento, de menor a mayor dificultad y revisión del mismo, -ir hacia atrás, siempre que sea necesario-. Por ello, son alternativas ideales para esta dificultad.

A pesar de ser diseñados como reemplazos a una tutoría presencial, un enfoque más constructivo es aplicable a este escenario: complementan el aprendizaje impartido en el aula y lo traslada a la virtualidad ofreciéndole una variedad de

² Colabora.NET: Conceptos fundamentales de la POO (Programación orientada a objetos) – Versión Electrónica <En Línea> URL: http://www.elguille.info/colabora/puntoNET/canchala_FundamentosPOO.htm

herramientas que facilitarán tal cometido y le permitirán vivenciar una experiencia más didáctica (que incluye explicaciones dinámicas, solución de ejercicios guiados, repasos), por tanto, el tutor inteligente podrá adaptarse a su aprehensión.³

1.2. FORMULACIÓN DEL PROBLEMA: EL SISTEMA TUTOR INTELIGENTE COMO COMPLEMENTO A LA PRESENCIALIDAD

El STI más que un reemplazo del docente, es un complemento de la enseñanza. Su funcionamiento radica en la adaptabilidad inherente que posee frente al conocimiento del alumno, pues es el primer paso en su proceso de instrucción: conocerlo y hacerle seguimiento a sus progresos para facilitar su estudio.

El conocimiento del alumno es un subconjunto del conocimiento del tutor, el cual se va incrementado a medida que se van alcanzando los hitos propuestos en la metodología a seguir.

Sin embargo, al implementarse en Web, el STI debe tener una serie de pasos metódicos para asistir al proceso de enseñanza-aprendizaje teniendo en cuenta que se rompe el esquema de la presencialidad, y que, en primera instancia no se cuenta con asistencia humana.

³ Urretavizcaya, Maite – Sistemas Inteligentes en el ámbito de la Educación. Versión Electrónica. <En Línea> URL: <http://aepia.dsic.upv.es/revista/numeros/12/Urretavizcaya.pdf>. Pág. 4.

Así pues, ¿qué metodologías y directrices se deben utilizar para la elaboración del diseño de un STI basado en Web, que sirva como complemento al proceso de enseñanza de la asignatura tratada?

2. JUSTIFICACIÓN

Bajo el enfoque constructivista que ilustra Urretavizcaya,⁴ el proyecto cobra importancia como una herramienta más en aras de mejorar el aprendizaje de los métodos numéricos. El STI permitirá que el estudiante se involucre más en la consecución de los objetivos de aprendizaje, al experimentar con los recursos que aquel le ofrece: explicación paso-a-paso de un ejercicio planteado, obtención de metas según un temario de contenidos.

Además, el STI le brindará al docente la posibilidad de medir la aprehensión de los estudiantes al ofrecerle un historial de cómo se han desenvuelto en aspectos como la proposición de un tema de estudio y la resolución de ejercicios planteados.

La libertad que ofrece, utilizar el sistema tutor desde cualquier posición geográfica a través de la Internet, es una gran ventaja para los actores involucrados: docente y alumnos; rompiendo así el paradigma del aula física. Esta premisa será ampliada más adelante.

La orientación de la idea es la adaptación del sistema al estudiante obteniendo en el proceso, toda la información posible sobre el mismo. La consecución de este

⁴ Urretavizcaya, Ibíd.

enunciado hace referencia al estudio del modelado del alumno, uno de los componentes de un STI.

Así pues, el problema del modelado del alumno se asocia a la representación de todos los procesos mentales que deben llevarse a cabo en el aprendizaje por computador.

El sistema tutor inteligente, objeto de esta investigación, será capaz de almacenar el desempeño de un estudiante y formular un diagnóstico de su estado de conocimiento. En cada problema, podrá, de ser necesario, anticiparse a una respuesta del alumno, simulando los procesos en los cuales un estudiante aprende y actúa según el problema que se le formule. Esto podrá alcanzarse fijando el conocimiento del estudiante como un subconjunto de todo el conocimiento que se desea enseñar (modelo de superposición), basado en la deducción de que la diferencia entre ambos conjuntos es una falta del conocimiento del alumno para obtener el total.

Continuando con el concepto de romper el esquema de la educación impartida en un aula física, el STI que se pretende implementar estará basado en Web, lo que le permitirá ser independiente del aula y de la plataforma en la cual se ejecute. Gracias a ello, el estudiante podrá recorrer los contenidos de una manera no secuencial dentro de lo que ha aprendido, ofreciéndole la posibilidad de retroceder a un contenido ya visto si cree que es necesario reforzar su conocimiento sobre el mismo. Esta característica del sistema, se conseguirá a través de la

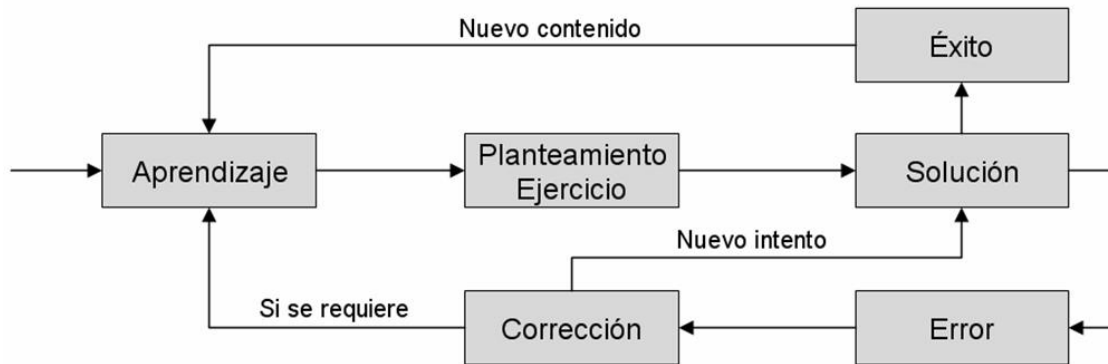
implementación de la hipermedia (multimedia e hipertexto), que facilitará la interactividad entre el alumno y el sistema y, posibilitará una futura adaptabilidad del mismo a los requerimientos de aprendizaje del alumno.

Mediante el sistema hipermedia, el estudiante podrá acceder a los contenidos de forma interactiva, y aprender de ellos. Por su parte el sistema estará en capacidad de asimilar la manera como el estudiante aprende y adaptarse de acuerdo a ello. El sistema también tendrá la capacidad de evaluar y retroalimentar al estudiante, ayudarlo a mejorar su conocimiento en las áreas en las que presenta debilidades y fortaleciendo las que mejor desempeño presenta, logrando mantener un conocimiento uniforme en todos los aspectos del contenido de estudio.

De acuerdo a esto, la organización de contenidos y problemas estará definida mediante una red semántica (grafo dirigido), en la cual el estudiante podrá ir explorando nuevos elementos gradualmente según la dificultad que vaya alcanzando. Este modelo se transforma en espiral porque el proceso es iterativo, pues ocurre de la misma manera por cada tópico que se quiera enseñar.

Así, la secuencia de estados del proceso de aprendizaje se puede representar en la siguiente gráfica (Ver Figura 1):

Figura 1. Estados de aprendizaje en un STI.



Tal sistema estará soportado en una arquitectura de cinco módulos consignados en la siguiente tabla:

Tabla 1. Módulos de un STI.

Módulo	Función
Modelo del dominio.	Conocimiento y metodología a seguir.
Modelo del alumno.	Seguimiento y traza del aprendizaje del alumno.
Modelo del tutor.	Asignación de actividades; acompañamiento del aprendizaje.
Modelo pedagógico	Toma de decisiones.
Interfaz.	Interacción entre el sistema y el usuario final.

2.1. OBJETO

Diseñar un sistema tutor inteligente que complemente la enseñanza de los métodos numéricos para la solución de ecuaciones algebraicas no lineales, mediante una enseñanza basada en el acompañamiento y seguimiento al estudiante y bajo una plataforma Web que permita su uso sin importar la ubicación geográfica.

3. OBJETIVOS

3.1. OBJETIVO GENERAL

Diseñar un sistema tutor inteligente (STI) basado en Web aplicado a la resolución de ecuaciones algebraicas no lineales utilizando métodos numéricos para ingeniería.

3.2. OBJETIVOS ESPECÍFICOS

1. Determinar los requerimientos del STI de tal forma que sean claros y comprensibles, identificando los actores que interactúan en el proceso y los escenarios de desarrollo, reconociendo su aplicación sobre el dominio de aprendizaje que se pretende modelar.
2. Diseñar un modelo de obtención de conocimiento (red semántica) que formule la estructura del mismo, defina la consecución de los objetivos propuestos y que facilite la toma de decisiones del sistema frente a las respuestas del alumno.
3. Definir el nivel de conocimiento del alumno para realizar estrategias que permitan la mayor comprensión del dominio de aprendizaje.

4. Diseñar el modelo representativo de cada actor que participa en el proceso: maestro, estudiante, dominio de aprendizaje; y la interfaz de comunicación entre cada uno de estos módulos.
5. Establecer metodologías de diseño y abstracción del STI para los requerimientos obtenidos anteriormente utilizando el Lenguaje Unificado de Modelado (UML).
6. Diseñar una arquitectura que permita el uso del STI bajo la Web, usando una comunicación cliente-servidor.

4. ALCANCE

Mediante esta investigación se pretende diseñar un sistema tutor inteligente basado en Web, capaz de complementar en la enseñanza de la resolución de problemas matemáticos asociados con ecuaciones algebraicas no lineales utilizando métodos numéricos para ingeniería. El STI podrá servir de apoyo dentro de la asignatura impartida como un asistente extracurricular que afiance los contenidos explicados.

El área de estudio, marco de acción de la investigación, serán los sistemas tutores y su adaptación al usuario (estudiante) y al entorno (Web), las estrategias de aprendizaje que el sistema inherentemente propondrá a un alumno en el planteamiento de un contenido, la solución de ejercicios propuestos y la retroalimentación que el sistema deberá realizar al alumno para describir su desempeño.

Inicialmente, sólo se abarcará el capítulo que trata sobre el estudio de los problemas algebraicos no lineales, como un temario de experimentación. Se entiende que los contenidos de la asignatura son muy amplios y denotan un gran esfuerzo de estudio. No obstante, el propósito de los autores es dejar sentadas las bases de un sistema lo suficientemente flexible para adaptarse a cualquier

asignatura pues los pasos en el aprendizaje son siempre los mismos: Explicación del problema, estrategias de solución, planteamiento de problemas que corroboren lo aprendido y retroalimentación.

Este esquema –cíclico- de aprendizaje depende en gran proporción de la aprehensión del estudiante, que a su vez modelará el modo de enseñanza por parte del tutor.

5. MARCO TEÓRICO

5.1. INFORMÁTICA EDUCATIVA

En el proceso educativo de la humanidad siempre se ha buscado la participación de agentes colaborativos que posibiliten la consecución de su objetivo: la propagación y generación del saber.

El cometido de la educación, voluntario y metódico, puede ser llevado a cabo de varias maneras. Se habla de educación formal –como la que se imparte dentro de un espacio físico-, educación no formal –la que ocurre fuera del anterior sistema-, y educación informal –consecuencias educativas no elaboradas específicamente con fines educativos-.⁵

En todos los casos, especialmente en el de la educación formal, se pretende bajo el concepto mismo de la educación, que una persona enseñe y otra aprenda. En consecuencia, impartir enseñanza y aprender se asocia a un principio: la influencia de quien enseña y la posibilidad de influenciarse por parte de quien aprende.⁶

⁵ Impacto de la informática en la educación. Educación – Ciberhábitat, Ciudad de la Informática – Versión Electrónica <En Línea> URL: <http://ciberhabitat.gob.mx/universidad/ui/eadei/eadeii.htm>

⁶ Sarramona, J. en Impacto de la informática en la educación. Educación – Ciberhábitat, Ciudad de la Informática – Versión Electrónica <En Línea> URL: <http://ciberhabitat.gob.mx/universidad/ui/eadei/eadeii.htm>

Manteniendo esta premisa, lo que puede llegar a cambiar es la forma, no el fondo. La educación a medida que transcurre el tiempo, se vale de herramientas para que su proceso se cumpla cada vez de mejor manera, para reforzar contenidos a quienes ya disfrutaban de ella, y llegar más lejos a brindarse a aquellos que no la poseen.

Las nuevas tecnologías de la información, frutos de la creatividad humana para hacer mejor su existencia, cobran protagonismo en esta tarea al facilitarle al ser humano tal objetivo, rompiendo esquemas mentales como la presencialidad y la aglomeración.

Su principal ventaja, ha sido la de educar a base de imágenes, dejando al texto escrito en un plano no muy inferior. La diferencia radica en un aprendizaje a base de movilizar emociones, sentimientos y la afectividad, al contrario de la lectura, que es reflexiva y racional.⁷

La tecnología aplicada a la educación está orientada a racionalizar y mejorar los procesos educativos mediante la sistematización de esfuerzos y aplicación de principios de ciencias de la educación.⁸

La interactividad, concepto inherente de la educación, sin embargo, no es aplicable a todas las tecnologías, pues esta saca provecho de la inteligencia del receptor invocando una decisión continua.⁹

⁷ Impacto de la informática en la educación. Tecnología – Ciberhábitat, Ciudad de la Informática – Versión Electrónica <En Línea> URL: <http://ciberhabitat.gob.mx/universidad/ui/eadei/eadeiv.htm>

⁸ Universidad de Deusto. Educación. Una incógnita en el cambio actual. Bilbao, 1998. Pág. 154.

⁹ Sánchez Ilabaca, Jaime – Aprender Interactivamente con los Computadores – Versión Electrónica <En Línea> URL: <http://www.c5.cl/ie/ie/articulos/aprender.html>

Por ejemplo, si se aplica tal precepto a los medios de comunicación, como la radio o la televisión, éste se fragmenta pues quien aprende no tiene oportunidad de refutar o cuestionar aquello que recibe: simplemente lo toma.

Dentro de estas tecnologías se destaca la informática, a partir del surgimiento de la computadora como instrumento potencializador de trabajo y con la posibilidad de ofrecer una gama de recursos visuales que pueden captar la atención de quien aprende, y con ellos establecer un diálogo con las respuestas dadas de parte y parte.

Se habla de la informática como la ciencia que maneja y estudia el tratamiento automático de la información mediante el uso de la computadora.¹⁰

En el caso de la educación, la incursión de esta disciplina se ha confundido con la instalación de computadoras para la administración de recursos en la escuela (administración escolar informatizada), con la enseñanza de la computación (enseñanza de la informática), o a la instalación de recursos computacionales para la investigación y la búsqueda de la información (acceso a recursos informáticos de información).¹¹

La aplicación de la informática a la educación, o la informática educativa implica el uso de las tecnologías de la información y la comunicación con intencionalidad

¹⁰ Cute Paste – Glosario. Versión Electrónica. <En Línea> URL: <http://cutepaste.wordpress.com/glosario-y-definiciones/>

¹¹ De Llano, José Gregorio, Adrián, Mariela - La Informática Educativa en la Escuela. Fe y Alegría – Fundación Santamaría. Caracas, 1998. Pág. 24.

pedagógica, integrándolas como recursos dentro de la planificación del proceso de aprendizaje.

Nunca se sustituirá la labor del profesor porque la informática es una herramienta con la cual se puede operar el proceso educativo; pero el diseño previo, la planificación y la estrategia a utilizar estará a cargo del experto educativo y del profesional (en ocasiones son la misma persona aunque no siempre es así).¹²

5.2. INTELIGENCIA ARTIFICIAL

5.2.1. DEFINICIÓN

Una de las metas de la ciencia, ha sido simular el comportamiento de los seres humanos. Gracias a esto, se han desarrollado diversas disciplinas que investigan cómo poder simular, no sólo el comportamiento, sino también la forma en cómo la mente humana procesa la información que recibe a través de los sentidos, cómo ésta puede tomar una serie de conceptos, situaciones, evidencias, etc., para analizarlas, aprender de ellas, tomar decisiones, etc.

¹² Impacto de la informática en la educación. Informática educativa – Ciberhábitat, Ciudad de la Informática – Versión Electrónica <En Línea> URL: <http://ciberhabitat.gob.mx/universidad/ui/eadei/eadev.htm>

Así pues, con la invención de la informática, la posibilidad de simular la mente humana toma un nuevo camino. Gracias a los computadores, se comienzan a desarrollar programas, o métodos, que intentan emular la inteligencia propia de los seres humanos.

Se entiende entonces, que la Inteligencia Artificial, es una disciplina de la ciencia que busca estudiar y simular la inteligencia que presenta la mente humana, apoyándose en la informática como herramienta de trabajo.¹³

5.2.2. HISTORIA

Crear una máquina capaz de pensar por si misma, ha sido uno de los sueños de los seres humanos desde la evolución de la ciencia a como se conoce hoy. Pero el estudio de la inteligencia de las maquinas tomó fuerzas gracias al matemático Alan Turing (1912-1953), quién en el año 1936 publicó el artículo “*On Computable Numbers*” (Sobre números calculables). En este, Turing estableció las bases teóricas para las ciencias de la computación.¹⁴ Presentó en él la conocida Máquina

¹³ Parra Castrillón, Eucario - Sistemas Tutoriales Inteligentes, Un Aporte de la Inteligencia Artificial para la Mediación Pedagógica - Versión Electrónica <En Línea> URL: <http://www.alfa.une.edu.ve/biblio/BiblioGenrerar/S/S/sistemastutoriales.asp>

¹⁴ Turing, Alan M. On Computable Numbers, with an application to Entscheidungsproblem. 1936. Versión Electrónica <En Línea> URL: <http://www.abelard.org/turpap2/tp2-ie.asp>

de Turing, la cual es un sistema teórico que lee una serie de instrucciones para así ejecutar una serie de operaciones con estas instrucciones.

Turing es conocido como el padre de la inteligencia artificial gracias al artículo *Computing Machinery and Intelligence*, que publicó en el año 1950, en el cual propuso la manera de cómo determinar si una máquina podía ser inteligente o no. El método fue llamado la Prueba de Turing.¹⁵

Aun así, la inteligencia artificial no era conocida como tal. Esto cambió en el año de 1956 en la Conferencia de Dartmouth, en la cual se hicieron previsiones triunfalistas a diez años que jamás se cumplieron, lo que provocó el abandono casi total de las investigaciones durante quince años.¹⁶

También, se llegó a la definición de las presuposiciones básicas del núcleo teórico de la inteligencia artificial:

1. El reconocimiento de que el pensamiento puede ocurrir fuera del cerebro, es decir, en máquinas.
2. La presuposición de que el pensamiento puede ser comprendido de manera formal y científica.

¹⁵ Turing, Alan M. *Computing Machinery and Intelligence*. 1950. Versión Electrónica <En Línea> URL: <http://www.abelard.org/turpap/turpap.php>

¹⁶ Inteligencia Artificial. Versión Electrónica <En Línea> URL: http://www.inteligenciaartificial.cl/ciencia/software/ia/inteligencia_artificial.htm

3. La presuposición de que la mejor forma de entenderlo es a través de computadoras digitales.¹⁷

Se ve que a medida cómo avanzaban los estudios en este campo, se van presentando diversas ramas de estudio. Algunos investigadores se dedicaron a estudiar los procesos de reconocimiento visual de patrones, al estudio de la naturaleza del aprendizaje en las computadoras.

Basándose en los estudios sobre memoria asociativa, el equipo Newell-Shaw-Simon construyó los primeros lenguajes de procesamiento de información (IPL-I, IPL-II) utilizados en el diseño de su "*Logic Theorist Machine*" que se convirtió en la primera máquina "inteligente".¹⁸

En los años sesenta, McCarthy y Raphael iniciaron el trabajo de diseñar y construir un robot móvil, al cual llamaron *Shakey*. Su objetivo era que *Shakey* pudiera interactuar con el entorno del mundo real. Este reto inició una fuerte preocupación en la IA por el estudio de la epistemología y los procesos cognoscitivos. La discusión se centró alrededor de los problemas de la representación mental o interna del conocimiento, la percepción y los problemas

¹⁷ Elguea, Javier. Inteligencia artificial y psicología: la concepción contemporánea de la mente humana. Versión Electrónica. <En Línea> URL: http://biblioteca.itam.mx/estudios/estudio/estudio10/sec_16.html

¹⁸ Ibíd.

del significado. La idea básica de Raphael era la de reunir, en una sola, distintas máquinas con capacidad de aprender por experiencia, de reconocer patrones visuales, de modelar, de manipular símbolos, etc., y esperar que el todo fuera mayor que la suma de las partes.¹⁹

Con este trabajo, mostró que construir una máquina capaz de funcionar con altos niveles de incertidumbre, como funciona la mente humana, era uno de los retos que la inteligencia artificial tendría que resolver. La inteligencia artificial, se vio influenciada por diversas disciplinas, tales como la lógica, la psicología, las matemáticas.

Uno de los enfoques que toma la inteligencia artificial, fue la del lenguaje, buscando tener herramientas capaces de entender el lenguaje y poder traducirlo a otros idiomas, y para esto la máquina tendría que entender el texto ingresado antes de proseguir a traducirlo.

Con esta idea se desarrollaron varios programas que se enfocaban en el lenguaje y la comprensión. Algunos de estos son: el "oráculo" de Phillips, "Sad Sam" de Lindzay, y "Eliza" de Wizenbaum y Colby.²⁰

¹⁹ Ibíd.

²⁰ Ibíd.

La inteligencia artificial no sólo se involucró en el lenguaje y su comprensión, también exploró el problema de la representación mental del conocimiento.

A mediados de los sesentas, en Stanford, se desarrolló una aplicación que hacía espectogramas y diagnósticos médicos, llamado DENDRAL.²¹

En los años 80, en Japón se presentó el desafío de la quinta generación de computadoras, esto llevó a la aparición de los sistemas expertos, los cuales no se desarrollaron satisfactoriamente, por lo cual su investigación se vio interrumpida en los años 90.

En los comienzos de los años noventa, un nuevo paradigma aparece llamado Agentes Inteligentes, los cuales son sistemas que perciben el entorno y toman acciones que maximizan las posibilidades de sucesos.²² Los agentes van desde programas que resuelven problemas específicos, buscando que los más complejos sean capaces razonar como un ser humano.

²¹ Santana Tejero, Ruth. DENDRAL: El primer sistema experto basado en conocimiento. Versión Electrónica <En Línea> URL: <http://www.it.uc3m.es/jvillena/irc/practicas/estudios/DENDRAL.pdf> Pág. 3.

²² Botti, Julián. Agentes Inteligentes: el siguiente paso en la inteligencia artificial. Versión Electrónica <En Línea> URL: <http://www.ati.es/novatica/2000/145/vjulia-145.pdf> Pág. 1.

A finales de esta década, Martin Fischles y Oscar Firschein describieron los atributos de un agente inteligente.²³ Dichos atributos del agente inteligente son:

1. Tiene actitudes mentales tales como creencias e intenciones.
2. Tiene la capacidad de obtener conocimiento, es decir, aprender.
3. Puede resolver problemas, incluso particionando problemas complejos en otros más simples.
4. Entiende. Posee la capacidad de crearle sentido, si es posible, a ideas ambiguas o contradictorias.
5. Planifica, predice consecuencias, evalúa alternativas (como en los juegos de ajedrez)
6. Conoce los límites de sus propias habilidades y conocimientos.
7. Puede distinguir a pesar de las similitudes de las situaciones.
8. Puede ser original, creando incluso nuevos conceptos o ideas, y hasta utilizando analogías.
9. Puede generalizar.
10. Puede percibir y modelar el mundo exterior.
11. Puede entender y utilizar el lenguaje y sus símbolos.

²³ Loncán, Pierre. Estructura de la Información. Versión Electrónica <En Línea> URL: http://cursa.ihmc.us/servlet/SBReadResourceServlet?rid=1201385077671_1590506156_7659

5.2.3. APLICACIÓN EN LA EDUCACIÓN

Con la evolución de los computadores, buscando formas de mejorar la calidad de vida de los seres humanos, la educación toma su parte en las ciencias de la computación. Buscando la manera de enseñar lo que normalmente un docente haría a un estudiante, se comienzan los estudios de cómo utilizar la computación en el trabajo de guiar a un estudiante en algún tema.

Con esta idea, es perceptible que los computadores han sido usados por más de 20 años como herramienta educativa. Se ve entonces la aparición de programas que ayudan en la tarea de la educación. Inicialmente estas aplicaciones fueron construidas con el fin de encontrar una solución a un problema específico. A medida que las investigaciones continuaron, se comenzaron a desarrollar aplicaciones que buscaban ayudar a un individuo a resolver diversos problemas.

Estos sistemas, llamados Computer-based Training (CBT), Entrenamiento basado en Computadoras, y Computer Aided Instruction (CAI), Instrucciones guiadas por Computadoras. Estos sistemas presentaban una serie de problemas con el fin de mejorar el desempeño de un estudiante en un área específica. Sin embargo, estos

sistemas no eran individualizados, estaban dirigidos a estudiantes en general, son había individualidad.²⁴

Debido a que en estos sistemas la individualidad de un estudiante no existía, los estudios continuaron buscando mejoras en este campo. Esto llevó a ampliar las investigaciones en este campo, y que éstas fueran tomando nuevos caminos. La inteligencia artificial entró en juego en este punto, y con esto nuevos sistemas fueron desarrollados. Éstos sistemas son conocidos como Sistemas Tutores Inteligentes (STI), *Intelligent Tutoring Systems* (ITS's).

Los STI, ofrecen una mayor flexibilidad de la presentación de contenidos y una mejora en las respuestas a los estudiantes.

Los STI, buscando cómo encontrar una manera de enfocar la inteligencia artificial a la educación, se apoyaron en disciplinas como la psicología, la pedagogía, etc., con el objetivo de modelar la toma de decisiones que un docente humano haría de acuerdo al entorno que lo rodea, tomando en cuenta cómo un alumno se desempeña y de cómo no la misma metodología de enseñanza consigue alcanzar que un estudiante comprenda el tema tratado.²⁵

²⁴ Bloom, B.S. 1984. The 2 Sigma Problem: The Search for Methods of Group Instruction as Effective as One-to-One Tutoring. *Educational Researcher*, 13, pp. 3-16. en *Applications of AI in Education – Crossroads: The ACM Student Magazine – Versión Electrónica* <En Línea> URL: <http://www.acm.org/crossroads/xrds3-1/aied.html>

²⁵ *Inteligência Artificial na Educação – Versión Electrónica* <En Línea> URL: <http://www.nce.ufrj.br/ginape/publicacoes/trabalhos/RenatoMaterial/iaeducacao.htm>

5.3. SISTEMAS TUTORES INTELIGENTES

5.3.1. DEFINICIÓN

Un tutor inteligente (STI) es un sistema de software que utiliza técnicas de inteligencia artificial para representar el conocimiento e interactúa con los estudiantes para enseñárselo.²⁶ No obstante, es necesario considerar los posibles estilos cognitivos que cada estudiante pueda tener.

Los STI surgen como evolución de los CBT's (Computer Based Training) – Entrenamiento basado en Computadoras- que a su vez nacieron a finales de los años 70's en universidades de EEUU como intentos de simular la lógica humana. La evolución a STI's consistió en la adición de la adaptabilidad que podía tener frente a estilos de aprendizaje diferentes.

Un STI se caracteriza por ser:

- Inteligente, pues tiene la habilidad de decidir qué, cómo y cuándo estudiar emulando a un profesor real.

²⁶ Van Lehn en Fundamentos para una metodología de diseño de sistemas tutoriales inteligentes centrada en la reparación de mecanismos – Sierra, García-Martínez y Cataldi – Versión Electrónica <En Línea> URL: <http://www.itba.edu.ar/capis/webcapis/RGMITBA/comunicacionesrgm/cacic2004-metodologia-diseno-sistemas-tutoriales-inteligentes.pdf>. Pág. 1.

- Flexible, pues permite una amplia gama de posibilidades de navegación para los estudiantes.
- Interactivo, porque sus canales de comunicación permiten el cruce de ideas entre el mismo y el estudiante y viceversa.
- Adaptativo, porque puede cambiar las estrategias de enseñanza en el transcurso del proceso enseñanza-aprendizaje.²⁷

El dominio de un STI abarca capacidades de entendimiento, aprendizaje, razonamiento y resolución de problemas.²⁸ De igual manera, debe identificar fortalezas y debilidades de un estudiante así como adaptar las estrategias de enseñanza según se requiera.

De esta manera, un STI está a la vanguardia de una aplicación de software educativo puesto que éste está dirigido a un dominio específico o una necesidad particular; quien lo usa está propensa a ser autodidacta, pues no tiene a alguien que lo guíe y corrija a lo largo del proceso. Un STI, se basa en cambio, en una serie de instrucciones y reglas generales para instruir a través de tareas de enseñanza.

²⁷ Parra Castrillón, Eucario - Op. Cit.

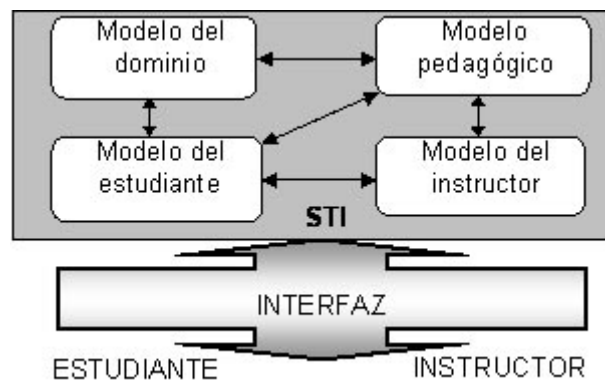
²⁸ Huapaya, C. R., Arona, G. M. y Lizarralde, F. A. - Enseñanza de la Ingeniería con Sistemas Tutoriales Inteligentes - Versión Electrónica <En Línea> URL: http://www.scielo.cl/scielo.php?pid=S0718-07642005000500012&script=sci_arttext&tIng=es.

Un STI no puede ser diseñado como una serie de instrucciones algorítmicas, porque su objetivo es el de representar el razonamiento humano, lo cual no puede ser alcanzado así. Deben aplicarse un conjunto de reglas de inferencia para lograr tal cometido.

5.3.2. COMPONENTES DE UN STI

Una arquitectura típica de una STI consta de cuatro componentes, que interactúan entre sí y de la siguiente manera:

Figura 2. Arquitectura de un Sistema Tutor Inteligente²⁹



El modelo del estudiante o del alumno contiene la información asociada al usuario final, tales como los aspectos psicológicos y sociológicos que condicionan o

²⁹ *Ibíd.*

limitan la enseñanza, el conocimiento a priori que el alumno posee sobre el tema de aplicación del STI, las habilidades sobre el tema. Es responsable de la traza de seguimiento del estudiante, su comportamiento y evolución dentro del proceso de aprendizaje. Toda esta información será la base de la adaptación de la metodología de enseñanza. Es responsable de abstraer el proceso mental del estudiante en su instrucción.

El modelo del dominio, almacena todo el conocimiento que se quiere impartir, el cual debe ser específico y detallado. Contiene la representación del mismo asociado a los procesos evaluativos, de enseñanza y de metodología, la cual dirige según el avance del alumno.

El modelo del tutor o modelo pedagógico es quién hace las veces de profesor o tutor. Es el responsable de tomar decisiones de acuerdo a los objetivos propuestos en la metodología del curso. Tendrá también la capacidad de evaluar y corregir y determinar la metodología a utilizar. También debe ser capaz de distinguir a cada tipo de estudiante y cuál es el nivel de ayuda que puede ofrecerle. Es responsable de activar la Interfaz para ilustrar el conocimiento. La interfaz permite la interacción entre el estudiante y el sistema. Es la única vía de acceso físico al avance del estudiante conforme transcurre el proceso. Debe ser flexible, tanto como se pueda, dotada de material atractivo y permisible a la comprensión del usuario. La multimedia es un aspecto muy importante a tener en

cuenta al momento de diseñar. El éxito de un programa educativo depende en gran parte de la riqueza comunicadora que reúna.³⁰

- **Modelo del dominio**

El dominio del conocimiento está recopilado en esta abstracción del sistema tutor. Un modelo del dominio debe satisfacer dos propósitos: presentar la materia de forma adecuada para que el alumno adquiriera las habilidades y conceptos, incluyendo la posibilidad de generar preguntas, plantear problemas, ofrecer explicaciones cuando se requiera. Por otro lado, debe ser capaz de resolver los problemas generados y corregir las soluciones presentadas y aceptarlas inclusive si se ha obtenido una solución por un método no esperado o no conocido. Debe poder explicar sus razonamientos de tal forma que sean entendibles para el alumno.³¹

Es importante conocer qué tipo de conocimiento es el que se esta intentando modelar. Se habla de conocimiento procedimental cuando es propio de cada dominio en particular. Este tipo de conocimiento tiene una base complementada con un conjunto de reglas. Es bastante similar al conocimiento básico de un sistema experto.

³⁰ González, Carina S. - Sistemas Inteligentes en la Educación: Una revisión de las líneas de investigación y aplicaciones actuales - Versión Electrónica <En Línea> URL: http://www.uv.es/RELIEVE/v10n1/RELIEVEv10n1_1.pdf., Pág. 4.

³¹ Millán, Eva. Sistema Bayesiano para el Modelado del alumno. Universidad de Malaga. Tesis doctoral. Versión Electrónica. <En Línea> URL: www.lcc.uma.es/~eva/investigacion/SBMA.pdf Pág. 10.

El conocimiento de tipo declarativo consiste en hechos organizados sistemáticamente para razonar directamente sobre ellos.

Los conocimientos de tipo cualitativos son los más difíciles de modelar, puesto que se tratan de relaciones espaciales y procesos dinámicos. En otras palabras tratan de modelar razonamientos causales.

Se describe tres tipos de modelos del dominio: ³²

- Caja negra: Tienen un comportamiento de entrada y salida sobre un rango de tareas sobre el dominio de conocimiento y pueden ser utilizadas como base para realizar correcciones. Este método se fundamenta en una base de ejercicios que soluciona y que considera como patrones de comparación con las posibles soluciones que un alumno pueda dar. Ofreciendo una retroalimentación constante e indicando según su solución propia si la que dio el alumno es o no correcta.
- Caja de cristal: este modelo es muy similar a la intención humana de impartir conocimiento. Es producto de la ingeniería del conocimiento en la que un ingeniero y un dominio experto pueden identificar un problema y su alcance, numerar y formalizar los conceptos clave, formular un sistema e implementar el conocimiento. Inherente al sistema, está su capacidad de guiar en el conocimiento tal como lo hace un humano abarcando un amplio

³² Anderson, J. R. The expert module. Carnegie-Mellon University. 1988 – Versión Electrónica <En Línea> URL: <http://act-r.psy.cmu.edu/papers/78/Expert.Module.pdf> Pág. 22.

rango de comportamientos para intentar la solución de un problema, revisándolos e interpretándolos.

- **Cognitivos:** Es una especialización del modelo de caja de cristal, no obstante, para comportarse tal como lo hace un tutor humano, este debe desplegar su conocimiento de acuerdo a las restricciones que un humano haría, lo cual es una característica que a diferencia de su predecesor, implementa facilitando la comunicación con el alumno. Su principal objetivo es descomponer el conocimiento en partes significativas y usarlo de manera similar a como lo haría un humano. Sin embargo, este tipo de modelación requiere de un gran esfuerzo por la gran cantidad de limitantes y procesamiento, pues plantea la posibilidad de abstraer componentes psicológicos para modelar el aprendizaje y sacrificar otras para abaratar el costo en complejidad computacional.

- **Modelo del alumno**

El problema principal de un sistema tutor inteligente es la capacidad de adaptarse a un alumno: su manera de entender lo que se le enseña y cómo aplicarlo a la solución de ejercicios propuestos. Entonces, es necesario que se obtenga la mayor información posible sobre el mismo, lo cual se puede lograr en caliente, a medida que se van enseñando los objetivos de un curso, realizando diagnósticos constantes del estado de aprehensión del estudiante.

El componente encargado de realizar esta labor es el modelo del alumno que a su vez, debe tener la capacidad de abstraer el conocimiento del estudiante y definirlo a causa de ello.

El diagnóstico del estudiante debe generar una base de datos que recoja las habilidades e impericias del mismo durante el proceso de enseñanza-aprendizaje. Tal resultado obtenido a partir de la mayéutica (cuestionamiento persistente al estudiante para obtener una descripción o una serie de resultados), será el fundamento para el diseño de estrategias de aprendizaje o de adaptación al alumno.

Normalmente, el modelo del alumno debe comportarse como un simulador de los procesos mentales del estudiante, por lo que puede actuar predictivamente y anticiparse a una posible respuesta del estudiante o guiarle hacia una respuesta correcta en caso de un error.

Para tal fin se han empleado históricamente, varios tipos de modelado del estudiante:³³

- Superposición: El conocimiento del alumno es un subconjunto del conocimiento del tutor. La diferencia entre los conjuntos es considerada como falta de conocimiento del alumno. Este modelo funciona bien cuando el principal objetivo del tutor es transmitir conocimiento experto al estudiante. Este modelo presenta problemas de reacción cuando el alumno posee

³³ Millán, Eva. Op. Cit., Pág. 15.

conocimiento que desconoce el tutor, situación no considerada en el modelo.

- Diferencial: Una especialización del modelo de superposición, que a diferencia de aquel, si tiene en cuenta el conocimiento desconocido por el tutor pero lo consolida como una subcategoría de la total cognición del estudiante, reconociéndolo como la diferencia entre este y el tutor.
- Perturbación: Se diferencia del modelo de superposición ya que este se basa en el conocimiento incorrecto del alumno. Considera al alumno como potencialmente distinto al tutor en lo que respecta al conocimiento que ambos poseen, en cantidad y calidad. La técnica más aplicada para lograr este cometido es adicionarle al conocimiento experto los errores frecuentes de los estudiantes, para irlos detectando y coleccionando en una biblioteca en caso de nuevas apariciones.
- Basado en restricciones: El dominio de conocimiento es modelado como una serie de restricciones sobre los problemas. El modelo del alumno es detallado como la serie de restricciones que ha violado cuando ha intentado solucionar un problema. Tal modelo es reconocido por ser robusto y flexible. Su robustez radica en que no depende de la estrategia que haya seguido un estudiante. La flexibilidad se basa en que el sistema puede reconocer cualquier método de solución a un problema por más innovador o recursivo que haya sido.

- Traza del conocimiento: Se trata de determinar de qué es lo que sabe el estudiante, incluyendo el conocimiento correcto que tiene sobre el dominio como los errores que haya cometido al intentar resolver un problema de aplicación.
- Traza del modelo: Pretende analizar los métodos de solución de problemas. Es bastante útil cuando el objetivo es retroalimentar al estudiante, sobre todo en casos en los que necesite ayuda o ilustraciones para continuar.

- **Modelo del tutor**

El modelo del tutor contiene toda la base metodológica para el proceso de aprendizaje, es decir las estrategias y tácticas a seguir según el desempeño del alumno.³⁴ Por tanto siempre lleva cuenta del mismo realizando un diagnóstico de su estado del conocimiento.

Debe controlar las actividades e interacción instructora siguiendo varios niveles de control: En el nivel de control máximo, el sistema adapta sus acciones a las necesidades del alumno, llevando siempre el control. A nivel intermedio, el sistema y el alumno comparten el control a través de la generación de preguntas y respuestas entre ambos. En el nivel más bajo, la acción del sistema se reduce a modificar el entorno de aprendizaje, a causa de actividades tutorizadas.³⁵

³⁴ Jiménez Rey, Elizabeth, Grossi, Maria Delia y otros. Diseño de un Sistema Tutor Inteligente Multiagente. Revista Eletrônica de Sistemas de Informação, Edição 8, nº2 2006. Pág 4.

³⁵ Millán Eva. Op. Cit., Pág 12.

Se identifican dos tipos de tutores:

- Expositivos: Transmiten conocimiento sobre los hechos, y las inferencias van encaminadas a mantener el enfoque y la coherencia.
 - Procedimentales: Enseñan habilidades y procedimientos, y adicionalmente ordenan las habilidades que componen la habilidad principal que se desea que el alumno adquiera y elegir los ejercicios y ejemplos.
-
- **Interfaz**

El entorno de aprendizaje o interfaz da soporte a las actividades del alumno y a los métodos que se utilizan para realizarlas. Deben ser atractivos y fáciles de utilizar para minimizar la adaptación del alumno al sistema y que este se dedique completamente al proceso de aprendizaje.

Burton define las seis cuestiones clave en el diseño del entorno: ³⁶

- Los aspectos del dominio que se deseen representar.
- El nivel de abstracción de la representación.
- La fidelidad de la representación.
- El orden en la presentación de contenidos.
- Herramientas de corrección y ayuda.
- Nivel de control que ejerce la herramienta.

³⁶ Burton, R. en Millán, Eva. Op Cit. Pág. 13.

En la actualidad, a pesar de contar con herramientas cada vez más potentes para tal fin, estos aspectos son considerados preponderantes al momento de diseñar una interfaz.

La Internet y la aparición de la multimedia han facilitado este cometido integrándose en sistemas hipermedia y en sistemas tutores basados en Web.

5.3.3. FUNCIONAMIENTO DE UN STI

González en la Revista Electrónica de Investigación y Evaluación Educativa propone que el funcionamiento de un STI debe estar ligado a un plan de instrucciones³⁷ basado en una estructura a niveles de la siguiente manera:

- Unidades básicas de aprendizaje, cómo está organizado el conocimiento.
- Objetivos conceptuales, qué habilidades o competencias se quiere que el alumno obtenga a medida que estudia cada unidad básica.
- Procesos cognitivos, qué debe ocurrir en el estudiante para alcanzar los objetivos.
- Eventos instruccionales, cuáles son las condiciones adecuadas para el aprendizaje.

³⁷ Gonzalez, Carina. Op. Cit., Pág 6.

- Acciones instruccionales, qué acciones deben realizar el estudiante y el sistema a lo largo de una sesión de estudio.

Apoyado en esta estructura, la interacción de los componentes de la arquitectura se realiza bajo un proceso cíclico de la siguiente manera:

- Identificación del usuario por parte del sistema.
- El Modelo del tutor (MT) activa el Modelo del alumno (MA) para organizar una sesión acorde a las características del alumno identificado.
- MT construye el plan instruccional consultando el Modulo del dominio (MD) –qué conocimientos se impartirán- y el MA –conocimiento a priori del estudiante y cuál es el estilo que más le agrada para aprender-.
- MT dirigirá a la Interfaz (I) que mostrará y pondrá en ejecución todas las tareas que demande el plan. Las tareas pueden ser a manera de demostración, evaluación, refuerzo, etc.
- MT supervisa las actividades del alumno valiéndose del conocimiento almacenado en el MD.
- Si la evaluación no es satisfactoria –rendimiento insuficiente, nuevos contenidos requeridos- se reestructura el plan; de lo contrario se continúa con el mismo plan.

5.3.4. LOS SISTEMAS HIPERMEDIA ADAPTATIVOS.

- **El sistema hipermedia**

Un sistema hipermedia es aquel capaz de almacenar un documento y permitir la lectura de este en forma no convencional o no lineal.³⁸

Habiendo definido que es un sistema tutor inteligente, este por sí mismo, presenta la desventaja de presentar la tarea educacional de forma muy restringida a lo que el tutor decida o según esté diseñado.

Por si solo, un sistema hipermedia deja un campo de libertad muy amplio, lo que también es considerado una desventaja, pues carece de pautas a seguir durante el desarrollo de un tema, dejando al usuario una labor de autodidacta.

Sin embargo, un sistema hipermedia no es considerado como un instructor educacional, pues como ya se mencionó, no posee métodos para adiestrar ni para individualizar la educación.³⁹ También puede causar falta de atención o interés hacia el tópico sobre el que se esté informando.

A pesar de ello, un sistema hipermedia permite libertad de navegación para que cada quien decida el flujo de aprendizaje; integración de componentes visuales (audio, video, gráficos) para una presentación más enriquecedora y aumentar la usabilidad del sistema.

³⁸ Polson MC, Richardson JJ. En Fernández, Luisa y otros. Sistemas Hipermedia Adaptativos: una aproximación al tema. Revista Cubana de Informática Médica – CECAM. Versión Electrónica <En Línea> URL: http://www.cecam.sld.cu/pages/rcim/revista_5/articulos_htm/hipermedia.htm

³⁹ Ibíd.

La hipermedia permite, mediante relaciones y la organización no-secuencial de informaciones, una implementación sencilla de las estructuras conceptuales, generalmente de semántica compleja, de los dominios a enseñar.

No obstante, toda esa libertad puede provocar que el usuario se extravíe en el mar de información que se le presente y por otro lado no existe la posibilidad de evaluar la información presentada a modo de comprobar que el estudiante cumplió sus objetivos. Además, la información presentada no varía, aún si el alumno tiene dificultades para comprenderla; en otras palabras el sistema no se adapta al usuario.

- **Adaptación del sistema al usuario**

El sistema hipermedia adaptativo, como tal, es un sistema hipermedia capaz de ajustar su presentación y navegación a las diferencias de los usuarios que lo utilizan, reduciendo así los problemas de desorientación y falta de comprensión propios de los sistemas hipermedia no adaptativos.⁴⁰

La adaptación consiste en que el sistema construye un modelo con los objetivos, preferencias y el conocimiento de un usuario particular y usa esta información para adaptar la presentación de la información según sus características y necesidades.⁴¹

⁴⁰ Medina, Nuria. Taxonomía de Sistemas Hipermedia Adaptativos. Universidad de Granada. Versión Electrónica. <En Línea> URL: <http://lsi.ugr.es/~fguti/taller/06/Nuria%20Medina.pdf> Pág. 1.

⁴¹ González, Carina. Op. Cit. Pág 8.

Sin embargo, el diseño de un sistema hipermedia adaptativo debe responder a cuatro interrogantes: ¿Qué?, ¿Cómo?, ¿Cuándo?, ¿A qué adaptar?

Los sistemas hipermedia suelen diseñarse lo suficientemente abiertos para que encajen con cualquier dominio de conocimiento (SHA general), o dedicados específicamente a un área persiguiendo fines didácticos o educativos.

Tabla 2. Criterios de adaptabilidad en un Sistema Hipermedia Adaptativo⁴²

Criterio	¿Qué?	¿A Qué?	¿Cómo?	¿Cuándo?
Aplicabilidad del modelo	X			
Métodos de adaptación	X		X	
Objeto de la adaptación		X		
Tipo de prerrequisitos			X	
Integración de información	X			
Interacción usuario-adaptación			X	X
Creación de hiperdocumentos			X	X
Información contextual		X		

- **Métodos de adaptación**

Brusilovsky habla de dos tipos de adaptación.⁴³ La adaptación a la presentación, que se trata de mostrar la información pertinente, justa y necesaria, a cada usuario

⁴² Medina, Nuria. Op. Cit. Pág. 1.

según lo que requiera; la adaptación a la navegación, que consiste en que el sistema es quien decide que es posible ver, limitando su espacio de navegación para evitarle perderse en el entorno.

Medina, amplía cada método de adaptación en la siguiente tabla:

Tabla 3. Métodos de adaptación.⁴⁴

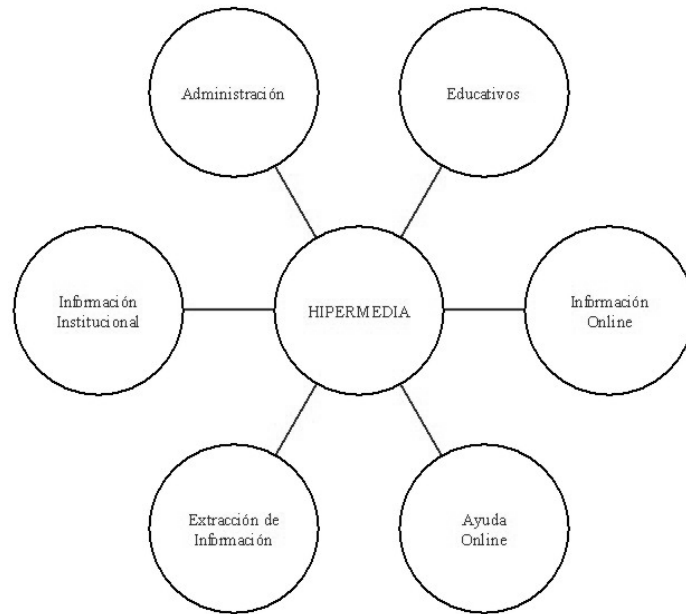
Adaptación de la Presentación	
Explicación Adicional	Proporcionar información adicional (ejemplos, ilustraciones, comentarios, etc.) a aquellos usuarios que la necesiten
Explicación de Prerrequisito	Incluir información sin la que el usuario no comprendería el resto de la página. Permite compensar la falta de un conocimiento requerido
Explicación Comparativa Variantes.	Incluir información sobre otros conceptos conocidos por el usuario que están relacionados con el concepto descrito en la página actual La misma información es presentada a cada usuario con distinto nivel de especialización, idioma, grado de verborrea, etc.
Ordenación	La información de la página es ordenada de acuerdo a algún criterio, que dependerá de la meta y otras características del usuario
Adaptación de la Navegación	
Consejo global	Sugerir un camino de navegación global: un conjunto de páginas y el orden de lectura
Consejo local	Sugerir la siguiente página a visitar
Soporte global de orientación	Mostrar una vista de la estructura de enlaces completa y la posición actual del usuario, indicando partes visitadas, deseables y prohibidas
Soporte local de orientación	Mostrar una parte de la estructura de enlaces, normalmente uno o dos niveles arriba o debajo de la página actual
Vistas personalizadas	Vista de la estructura de enlaces orientada a la meta del usuario

Los sistemas hipermedia necesitan ser adaptables según el área de aplicación. Se derivan seis clases de sistemas hipermedia reconocidos. (Ver Figura 3)

⁴³ Brusilovsky, Peter. Adaptive hypermedia: an attempt to analyse and generalize – International Centre for Scientific and Technical Information. Versión Electrónica. <En Línea> URL: <http://www.wis.win.tue.nl/ah94/Brusilovsky.html>

⁴⁴ Medina, Nuria. Op. Cit. Pág. 8.

Figura 3. Aplicaciones de los sistemas hipermedia adaptativos.⁴⁵



Los hipermedias educativos son pequeños hiperespacios que representan un curso o sección de un material de aprendizaje sobre un tema en particular dónde un alumno tiene como objetivo aprender este material o cierta parte del mismo, por lo tanto, la característica más relevante del usuario para el sistema es su nivel de conocimiento sobre este tema.⁴⁶

⁴⁵ González, Carina. Op. Cit. Pág. 9.

⁴⁶ Ibíd. Pág. 9.

5.4. MÉTODOS NUMÉRICOS PARA LA SOLUCIÓN DE ECUACIONES ALGEBRAICAS NO LINEALES

5.4.1. CONCEPTOS PRELIMINARES

- **Límite de una función**

Al calcular el límite de una función cuando tiende a un punto b , se hace un acercamiento al punto, pero nunca se alcanza. El valor de la función en ese punto es irrelevante. Es más significativo el valor de la función cuando tiende a dicho punto.

El límite de una función está definido de la siguiente manera:

Sea f una función definida, se dice que $\lim_{x \rightarrow b} f(x) = L$, si para cada número positivo ε , por pequeño que este sea, es posible determinar un número positivo δ , tal que para todos los valores de x , diferentes de b , que satisfacen la desigualdad $|x - b| < \delta$, se verificará la desigualdad $|f(x) - L| < \varepsilon$.⁴⁷

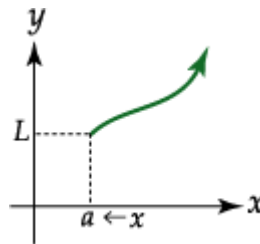
Es decir, $\lim_{x \rightarrow b} f(x) = L$ si y solo si para cada $\varepsilon > 0$ existe $\delta > 0$ tal que si $0 < |x - b| < \delta$, entonces $|f(x) - L| < \varepsilon$.

⁴⁷ Hernández S., Elsie. Cálculo Diferencial e Integral. Límites y Continuidad. Versión Electrónica <En Línea> URL: <http://www.cidse.itcr.ac.cr/cursos-linea/CALCULODIFERENCIAL/curso-elsie/limitesycontinuidad/html/limitesfinal.pdf> Pág. 10.

Esta definición de límite es principalmente usada para funciones continuas. Para funciones discontinuas, se recurre a los Límites Laterales.

- I. Límite por la derecha. Se dice que $\lim_{x \rightarrow a^+} f(x) = L$ si y solo si para cada $\varepsilon > 0$ existe $\delta > 0$ tal que si $0 < x - a < \delta$ entonces $|f(x) - L| < \varepsilon$. L es el límite por la derecha de $f(x)$ en a .⁴⁸

Figura 4. Límite por la derecha de una función.

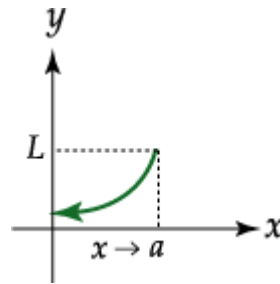


- II. Límite por la izquierda. Se dice que $\lim_{x \rightarrow a^-} f(x) = R$ si y solo si para cada $\varepsilon > 0$ existe $\delta > 0$ tal que si $0 < x - a < \delta$ entonces $|f(x) - R| < \varepsilon$. R es el límite por la izquierda de $f(x)$ en a .⁴⁹

⁴⁸ Ibíd. Pág 15.

⁴⁹ Ibíd. Pág 16.

Figura 5. Límite por la izquierda de una función.



Existen una serie de teoremas que permiten determinar el límite de una función.⁵⁰

Estos teoremas son:

i. **Unicidad del Límite:** Sea f una función definida en un intervalo $I \subset \mathbb{R}$ tal que $a \in I$. Si $\lim_{x \rightarrow a^-} f(x) = L$ y $\lim_{x \rightarrow a^-} f(x) = M$ entonces $L = M$.

ii. Si m y b son números reales entonces $\lim_{x \rightarrow a} (mx + b) = ma + b$

iii. Se cumple que $\lim_{x \rightarrow p} k f(x) = k \lim_{x \rightarrow p} f(x)$.

iv. Si $f(x) = \sqrt{x}$ con $x \geq 0$ entonces $\lim_{x \rightarrow a} \sqrt{x} = \sqrt{a}$, con $a \geq 0$.

v. Se cumple que $\lim_{x \rightarrow p} (f(x) + g(x)) = \lim_{x \rightarrow p} f(x) + \lim_{x \rightarrow p} g(x)$.

vi. Se cumple que $\lim_{x \rightarrow p} (f(x)g(x)) = \lim_{x \rightarrow p} f(x) \cdot \lim_{x \rightarrow p} g(x)$.

vii. Se cumple que $\lim_{x \rightarrow a} \frac{f(x)}{g(x)} = \frac{\lim_{x \rightarrow a} f(x)}{\lim_{x \rightarrow a} g(x)} = \frac{L}{M}$ siempre que $M \neq 0$

viii. $\lim_{x \rightarrow a} \frac{1}{x} = \frac{1}{a}$ siempre que $a \neq 0$.

⁵⁰ Ibíd. Pág. 19

ix. Si $n \in \mathbb{N}$ entonces $\lim_{x \rightarrow a} \sqrt[n]{x} = \sqrt[n]{a}$ si:

a. es cualquier número positivo.

b. $a \leq 0$ y n es impar.

x. Si $\lim_{x \rightarrow a} f(x) = L$, entonces $\lim_{x \rightarrow a} \sqrt[n]{f(x)} = \sqrt[n]{\lim_{x \rightarrow a} f(x)} = \sqrt[n]{L}$ si se cumple alguna de

las condiciones siguiente:

a. $L \geq 0$ y n es cualquier entero positivo ($n \in \mathbb{N}$).

b. $L \leq 0$ y n es un entero impar positivo.

xi. Si f , g y h son funciones tales que $f(x) \leq g(x) \leq h(x)$ para todo x de cierto

entorno reducido δ_1 de b y además $\lim_{x \rightarrow b} f(x) = L = \lim_{x \rightarrow b} h(x)$ entonces se cumple

que $\lim_{x \rightarrow b} g(x) = L$.

- **Continuidad de una función**

Inicialmente puede inferirse que una función es continua cuando esta al ser representada gráficamente, no presenta saltos, es decir, la gráfica se mantiene sin romperse en algún punto y continuar en otro. Ahora, matemáticamente para definir que una función es continua se establecen ciertas condiciones.

Se dice que una función f es continua en un punto c si y solo si:

i. $f(c)$ está definida, (es decir, c pertenece al dominio de f)

ii. $\lim_{x \rightarrow c} f(x)$ existe

iii. $\lim_{x \rightarrow c} f(x) = f(c)$

Si al menos una de estas condiciones no se cumplen, la función f es discontinua.⁵¹

Existen discontinuidades que pueden ser evitables. Cuando una función f es discontinua en un punto a , y a su vez el $\lim_{x \rightarrow a} f(x)$ existe, se presenta el caso que $f(a)$ no existe o que es diferente al $\lim_{x \rightarrow a} f(x)$.

Pero si el límite de la función en el punto de discontinuidad no existe, La discontinuidad será inevitable o esencial.

La continuidad de una función puede ser vista, no solo desde un punto, sino también desde un intervalo. Para que una función f sea continua en un intervalo $[a,b]$, se debe cumplir:

- i. f es continua para todo x tal que $x \in [a, b]$. Es decir, el $\lim_{x \rightarrow x_0} f(x) = f(x_0)$, para todo x_0 que pertenece al intervalo abierto (a,b) .
- ii. f es continua por la derecha en a . Es decir, $\lim_{x \rightarrow a^+} f(x) = f(a)$,
- iii. f es continua por la izquierda en b . Es decir, $\lim_{x \rightarrow b^-} f(x) = f(b)$

⁵¹ Ibíd. Pág. 72

- **Derivada de una función**

La derivada de una función en un punto indica el coeficiente de cambio de dicha función. Es decir, indica que tan rápido crece o decrece la función en un punto (la pendiente de la recta tangente en ese punto).

Ahora, se define que: La pendiente de la recta tangente a la curva con ecuación

$y = f(x)$ en el punto (x_0, y_0) , denotada $m_t(x_0)$ es igual al $\lim_{x \rightarrow x_0} \frac{f(x) - f(x_0)}{x - x_0}$, siempre que este límite exista.⁵²

Entonces, la derivada de una función se define:

Sea f una función real definida en un intervalo $I \subset \mathbb{R}$. Sea $x_0 \in I$

La derivada de f en el punto x_0 , denotada $f'(x_0)$, es el $\lim_{x \rightarrow x_0} \frac{f(x) - f(x_0)}{x - x_0}$ si este límite existe.

Si en la definición de derivada se sustituye $x-x_0$ por h , entonces $h \rightarrow 0$ cuando $x \rightarrow x_0$

$f'(x) = \lim_{h \rightarrow 0} \frac{f(x_0 + h) - f(x_0)}{h}$ y $x = x_0 + h$. Luego, si este límite existe. La función f es

⁵² Hernández S., Elsie. Calculo Diferencial e Integral. Derivada de una función. Versión Electrónica <En Línea> URL: <http://www.cidse.itcr.ac.cr/cursos-linea/CALCULODIFERENCIAL/curso-elsie/derivadafuncion/html/derivadas.pdf> Pág. 12.

derivable en x_0 si $f'(x_0)$ existe. Si $f'(x)$ existe para cada x en un intervalo I , ($I \subset \mathbb{R}$),

$$f'(x) = \lim_{h \rightarrow 0} \frac{f(x+h) - f(x)}{h}$$

se dice que la función f es derivable en I ; se escribe

53

Las notaciones para la derivada de una función con respecto a una variable, son:

$$D_x f, \frac{d}{dx} f(x), \frac{dy}{dx} \text{ (Notación de LEIBNITZ), } y'.$$

Si f es una función continua definida en $x=x_0$, se definen las derivadas laterales de la siguiente manera:

La derivada por la derecha, que se denota $f'_+(x_0)$, se define por la igualdad:

$$f'_+(x_0) = \lim_{x \rightarrow x_0^+} \frac{f(x) - f(x_0)}{x - x_0}, \text{ siempre que el límite exista. }^{54}$$

La derivada por la izquierda, denotada $f'_-(x_0)$, se define por la igualdad:

$$f'_-(x_0) = \lim_{x \rightarrow x_0^-} \frac{f(x) - f(x_0)}{x - x_0}, \text{ siempre que el límite exista. }^{55}$$

Para ayudar en el desarrollo de las derivadas, se han planteado una serie de teoremas, los cuales son:⁵⁶

- i. La derivada de una constante es igual a cero (0).

⁵³ Ibíd. Pág. 13.

⁵⁴ Ibíd. Pág. 16

⁵⁵ Ibíd. Pág. 16.

⁵⁶ Ibíd. Pág. 19.

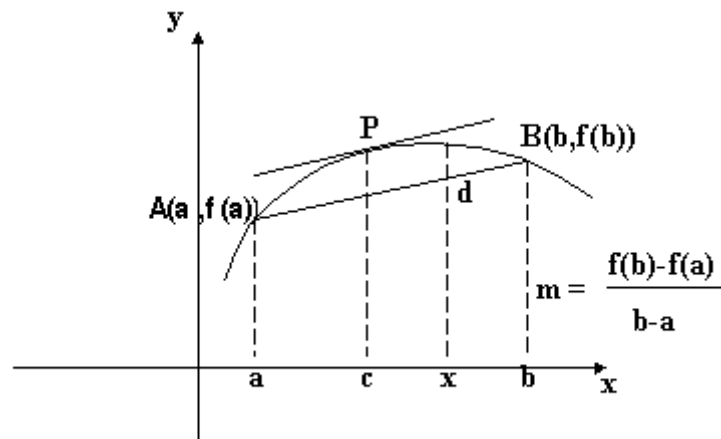
- ii. Si $f(x) = x$ entonces f es derivable sobre \mathbb{R} y $D_x f(x) = D_x x = 1$.
- iii. Si $f(x) = x^n$ con $n \in \mathbb{Q}$ y x pertenece al conjunto A en el que x^n está bien definida, entonces f es derivable en A y $D_x x^n = n x^{n-1}$.
- iv. Si la función f es derivable sobre un intervalo K y c es un número real, entonces la función g para la que $g(x) = c f(x)$ es derivable sobre K , además $D_x [c f(x)] = c D_x f(x)$.
- v. Si f y g son dos funciones derivables sobre un intervalo k , entonces la función $h = f + g$ es derivable sobre k y además $D_x [f(x) + g(x)] = D_x f(x) + D_x g(x)$, para $x \in k$.
- vi. Si f y g son funciones derivables sobre un intervalo k entonces la función $H = f \cdot g$ es derivable sobre k , y además para cualquier $x \in k$ se tiene que $D_x [f(x) \cdot g(x)] = f(x) D_x g(x) + g(x) D_x f(x)$.
- vii.
- Si f y g son dos funciones derivables y si $g(x) \neq 0$ sobre un intervalo k entonces la función $h = \frac{f}{g}$ es derivable sobre k , y además para cualquier $x \in k$ y se tiene que $D_x \left(\frac{f(x)}{g(x)} \right) = \frac{g(x) D_x f(x) - f(x) D_x g(x)}{[g(x)]^2}$.

- **Teorema del valor medio**

El Teorema del Valor Medio, conocido también como Teorema de los Incrementos Finitos o Teorema de Lagrange, plantea que en una función f cualquiera, continua en el intervalo cerrado $[a,b]$ y derivable en el intervalo abierto (a,b) , existe por lo menos un número c tal que:

$$f'(c) = \frac{f(b) - f(a)}{b - a}$$

Figura 6. Representación gráfica del teorema del valor medio.



El término $\frac{f(b) - f(a)}{b - a}$ es la pendiente de la recta secante a la curva que pasa por los puntos A y B . es la pendiente de la recta secante a la curva que pasa por los puntos A y B . De esta forma, se puede interpretar geoméricamente el teorema así: Existe un punto P sobre la curva de abscisa $c, c \in (a,b)$. tal que la recta

tangente a la curva en P cuya pendiente es $f'(c)$, es paralela a la recta secante \overline{AB} .⁵⁷

- **Teorema del valor intermedio**

Sea f continua en el intervalo $[a,b]$ y suponiendo que $f(a) < f(b)$. Entonces para cada z tal que $f(a) < z < f(b)$, existe un $x_0 \in (a,b)$ tal que $f(x_0) = z$.

En particular, si $f(a)$ y $f(b)$ tienen signos opuestos, entonces un valor intermedio es precisamente $z = 0$, y por lo tanto, el Teorema del Valor Intermedio permite asegurar que debe existir $x_0 \in (a,b)$ tal que $f(x_0) = 0$, es decir, debe haber *por lo menos* una raíz de $f(x)$ en el intervalo (a,b) .⁵⁸

- **Teorema de Taylor**

Sea f continua en $[a, b]$ y con derivadas hasta de orden n continuas también en este intervalo cerrado; supóngase que $f^{(n+1)}(x)$ existe en (a,b) , entonces para x y $x_0 \in (a,b)$ la aproximación de la función usando el polinomio de Taylor de grado n es:

⁵⁷ Del Valle Sierra, Jesús. Curso de Matemáticas dirigido a Estudiantes de Ingeniería de Sistemas. – Versión Electrónica <En Línea> URL: http://huitoto.udea.edu.co/Matematicas/Linea_Recta_Def.htm

⁵⁸ Tapia, Gustavo. Análisis Numérico – Versión Electrónica <En Línea> URL: <http://docentes.uacj.mx/gtapia/AN/Unidad2/iteracion.htm>

$$f(x) = f(x_0) + f'(x_0)(x - x_0) + \frac{f^{(2)}(x_0)}{2!}(x - x_0)^2 + \dots + \frac{f^{(n)}(x_0)}{n!}(x - x_0)^n + E_n$$

Donde $E_n = \frac{f^{(n+1)}(c)}{(n+1)!}(x - x_0)^{n+1}$ y c es un punto que se encuentra entre x y x_0 .

Para algunas funciones f , se puede probar E , se aproxima a cero cuando n se acerca al ∞ .⁵⁹

5.4.2. DEFINICIÓN DE MÉTODO NUMÉRICO

Según la definición que presenta Asmar en su libro 'Métodos Numéricos: Un primer curso', se entiende por método numérico el procedimiento por el cual se obtiene la solución de ciertos problemas realizando cálculos puramente aritméticos y lógicos. Estos procedimientos llevan a cabo una serie de instrucciones lógicas que producen una solución. La eficiencia del cálculo depende de la implementación y del instrumento que se utilice para el cálculo.⁶⁰

⁵⁹ Arenas, C. Fichas Matemáticas. – Versión Electrónica <En Línea> URL: <http://diposit.ub.edu/dspace/bitstream/2445/1322/1/218.pdf> Pág. 72.

⁶⁰ Asmar Ch., Iván – Métodos Numéricos. Un primer curso – Medellín: Universidad Nacional de Colombia Sede Medellín, Facultad de Ciencias, Departamento de Matemáticas, 1999, Pág. 3.

Sin embargo, la solución de los problemas mediante métodos numéricos suelen ser aproximadas, pues los métodos numéricos recalcan en la aplicación de modelos matemáticos o simplificaciones al problema real.⁶¹

Los parámetros que se pueden aplicar a un modelo matemático de cualquier tipo pudieron ser obtenidos de muestras experimentales lo que puede ocasionar errores que pueden clasificarse en errores inherentes, truncamiento o redondeo.⁶²

5.4.3. ERRORES

Un error es definido como la diferencia entre el valor real y el valor aproximado obtenido en la utilización de un modelo o método numérico, tal como lo muestra la siguiente ecuación:

$$e = V_R - V_A$$

Donde e es el error, V_R es el valor real y, V_A es el valor aproximado.

⁶¹ Leyva, Hugo Pablo - Métodos Numéricos. Teoría de Errores. <En Línea> URL: <http://sai.azc.uam.mx/apoyodidactico/mn/Unidad1/mntema1.html>

⁶² Ibíd.

Un error inherente es asociado a un parámetro del modelo, el cual se obtiene experimentalmente como se menciona antes, lo que ocasiona una pérdida de precisión en el resultado de la aplicación del método.

Un error de truncamiento se origina por aproximar una solución analítica mediante la aplicación de un método. Esto se produce en casos en los cuales es imposible tomar todos los posibles términos resultantes, por lo que es necesario truncarlos, lo que también origina aproximaciones al resultado exacto.

Por otro lado, los errores de redondeo son asociados a soluciones de tipo racional en los cuales no es posible tomar todos los números que hacen parte del residuo o conforman la parte decimal del resultado, lo cual conlleva a tomar sólo unos cuantos números de la misma, dependiendo de la solución requerida.

El error puede ser cuantificado de acuerdo a su magnitud. Entonces, un error se cuantifica porcentual o relativamente. El error relativo se obtiene como el cociente entre la diferencia del valor real y el valor aproximado y, el valor real, siempre y cuando el valor real sea diferente de cero. La siguiente ecuación lo ilustra:

$$e_R = \frac{e}{V_R} = \frac{V_R - V_A}{V_R}$$

El error porcentual, como lo indica su nombre es el error relativo representado en porcentaje, como lo indica la siguiente ecuación:

$$e_p = 100 \cdot e_R (\%) = 100 \cdot \frac{V_R - V_A}{V_R} (\%)$$

5.4.4. CIFRAS SIGNIFICATIVAS

Al obtener un error de redondeo, es importante tener en cuenta la aplicación de cifras significativas, para no alterar en demasía la solución aproximada y que esta pueda parecerse lo mayor posible a la solución real, para no obtener un error relativo muy alto. Por ejemplo, se obtiene como solución un número como 0.13745 y se solicita que ésta se entregada con dos cifras decimales. La aplicación de cifras significativas, permite redondear el resultado obtenido.

Leyendo la cifra de derecha a izquierda se puede incrementar el número siguiente si el número leído es mayor o igual que 5. Así pues, de 0.13745, puede obtenerse un número decimal de cuatro cifras como 0.1375. Del mismo modo se pueden despreciar cifras decimales si no aportan algo que permita incrementar la siguiente. Lo que para el ejemplo propuesto significa despreciar el último número sin que se afecte el resultado a 0.137. Aplicando esta mecánica, se puede llegar a 0.14 como solución aproximada y redondeada.

Como un método numérico obtiene una solución mediante la aplicación de una serie de pasos lógicos, es bastante difícil estimar cuan afectada por ello resulta; es decir cuánto se propaga. Dos criterios son útiles para determinar esta situación: la estabilidad y la convergencia.

5.4.5. ESTABILIDAD Y CONVERGENCIA

La estabilidad puede observarse en la aplicación de un método numérico cuando se entregan parámetros de entrada para obtener una salida de forma secuencial o algorítmica. Dependiendo de cuánto cambie la entrada, así podrá cambiar la salida. Si es muy notorio el cambio en la salida de un método, se dice que el método es inestable y que el problema está mal condicionado.

La convergencia se muestra en el resultado obtenido por cada iteración o nuevo paso del método empleado, formando una sucesión. Si cada término de la solución conduce de forma incremental o decremental a un número o resultado (límite) se dice que la solución es convergente. De otro modo es divergente.

5.4.6. TOLERANCIA

La sucesión de resultados se muestra de la siguiente forma:

$$x_1, x_2, x_3, \dots, x_n$$

La convergencia se determina matemáticamente como el límite hacia el cual tiende el n -ésimo término de la solución así:

$$\lim_{n \rightarrow \infty} x_n = x$$

Lo cual en la práctica es poco probable de obtener, por lo que se recurre a definir un criterio que determine la convergencia de la sucesión de valores. A veces, suele tomarse como este criterio alguno de los parámetros de cuantificación del error tales como el error relativo o porcentual. Pero si la convergencia existe, también debe existir el límite de los errores. Esto no es posible, porque en la mayoría de las veces que se emplea un método numérico no se conoce el valor real.

Lo mejor en estos casos es definir un tope o una tolerancia, lo suficientemente pequeño pero mayor que cero, de tal forma que en la obtención de una solución se establece hasta que el error relativo sea lo suficientemente cercano a la tolerancia. Para el caso de las sucesiones, el error relativo se calcula como la diferencia entre el valor anterior y el valor actual dividido entre el valor anterior.

Esto lo indica la siguiente ecuación:

$$e_R < \varepsilon \Leftrightarrow \left| \frac{V_n - V_{n-1}}{V_n} \right| < \varepsilon, V_n \neq 0$$

5.4.7. MÉTODOS CERRADOS

Los métodos numéricos que se emplean para obtener soluciones para una ecuación algebraica no lineal, suelen ser iterativos (repetitivos), y generan una sucesión de resultados, como se estableció anteriormente.

Un método para encontrar las raíces de una ecuación algebraica no lineal que en cada iteración encuentra un intervalo cerrado donde se encuentra la raíz, es un método cerrado.

No obstante, es necesario partir de una base para saber en que intervalo inicial tiene raíces la ecuación. Por lo que se plantea el siguiente teorema: Si en un intervalo cerrado $[a,b]$, la función $f(x)$ es continua y además $f(a)$ tiene signo opuesto al de $f(b)$, es decir, existe un cambio de signo (CS), entonces por lo menos existe una raíz en $[a,b]$ ⁶³. Este teorema es denominado el teorema de cambio de signo. Sin embargo, existen ocasiones en la que este teorema no se cumple para un intervalo y si existen raíces dentro de él.

⁶³ Ibíd.

- **Método de Bisección**

Dentro de los métodos cerrados, se distingue el método de bisección por el cual se divide un intervalo que contenga a la raíz tantas veces hasta que la longitud del mismo sea menor o igual que alguna tolerancia.

Como parámetros de entrada se entregan los extremos del intervalo y el punto medio entre ellos. Dado el intervalo cerrado $[a,b]$, entonces el punto medio x_1 entre ellos está dado por la siguiente ecuación.

$$x_1 = \frac{a+b}{2}$$

Tal punto medio representa la primera aproximación a la raíz de la ecuación.

El método termina cuando $f(x_1) = 0$ o $\frac{b_1 - a_1}{2} < \varepsilon \Rightarrow \alpha = x_1$, siendo ε la tolerancia dada para obtener la raíz α .

En caso de no alcanzarse las dos condiciones anteriores, es necesario aplicar el teorema de cambio de signo para reducir el intervalo y determinar los nuevos extremos del mismo.

Si $f(a) \cdot f(x_1) < 0$, entonces $b = x_1$, en caso contrario, $a = x_1$ y se vuelve a aplicar el método, esta vez con los nuevos valores.

El algoritmo del método de bisección⁶⁴ se muestra a continuación:

⁶⁴ Asmar Ch. – Op. Cit. Pág 39.

Entrada: $f(x)$, los extremos del intervalo $[a,b]$, una tolerancia ε , y un número máximo de iteraciones N .

Salida: Una raíz aproximada α^* o un mensaje.

Paso 1: Tomar $n = 1$

Paso 2: Mientras que $n \leq N$ seguir los pasos 3-6:

Paso 3: Tomar $x_n = \frac{a+b}{2}$

Paso 4: Si $f(x_n) = 0$ o $\frac{b-a}{2} < \varepsilon$, entonces **Salida:** $\alpha^* = x_n$. **Terminar.**

Paso 5: Tomar $n = n + 1$

Paso 6: Si $f(a) \cdot f(x_n) < 0$, entonces tomar $b = x_n$, de lo contrario $a = x_n$

Paso 7: **Salida:** Se alcanzó N pero no la tolerancia.

- **Método de Falsa Posición (Regula Falsi)**

El método de Regula Falsi (Regla Falsa en Latín), es también un método cerrado que requiere las mismas entradas que el método de bisección. Su mecánica es muy similar salvo que, a diferencia del método de bisección, no se busca el punto medio del intervalo como estrategia para dividir el intervalo sino la intersección de la recta entre los puntos $(a_n \cdot f(a_n))$ y $(b_n \cdot f(b_n))$ y el eje de las abscisas. Tal punto es la 'posición falsa' de la raíz de la ecuación. De ahí el nombre del método.

Este punto se obtiene después de la n -ésima iteración con la siguiente ecuación:

$$x_n = \frac{a_n \cdot f(b_n) - b_n \cdot f(a_n)}{f(b_n) - f(a_n)}$$

Siendo $n=1$, para la primera iteración, se calcula x_n con la ecuación anterior. Si $f(x_n)=0$ se ha obtenido la raíz de la ecuación, de lo contrario es necesario volver a iterar, pero para ello debe establecerse el nuevo intervalo donde se encuentra la raíz de la ecuación, para lo que debe emplearse nuevamente el teorema de cambio de signo. Si $f(a) \cdot f(x_n) < 0$, entonces $b = x_n$, en caso contrario, $a = x_n$ y se vuelve a aplicar el método, esta vez con los nuevos valores. Si se estableció una tolerancia ε o un máximo de repeticiones N , la aplicación del método se detiene cuando se haya alcanzado ε o N .

El algoritmo para el método de falsa posición es el siguiente:

Entrada: $f(x)$, los extremos del intervalo $[a,b]$, una tolerancia ε , y un número máximo de iteraciones N .

Salida: Una raíz aproximada α^* o un mensaje.

Paso 1: Tomar $n=1$

Paso 2: Mientras que $n \leq N$ seguir los pasos 3-6:

Paso 3: Tomar $x_n = \frac{a \cdot f(b) - b \cdot f(a)}{f(b) - f(a)}$

Paso 4: Si $f(x_n) = 0$ o $\left| \frac{x_n - x_{n-1}}{x_n} \right| < \varepsilon$, entonces **Salida:** $\alpha^* = x_n$. **Terminar.**

Paso 5: Tomar $n = n + 1$

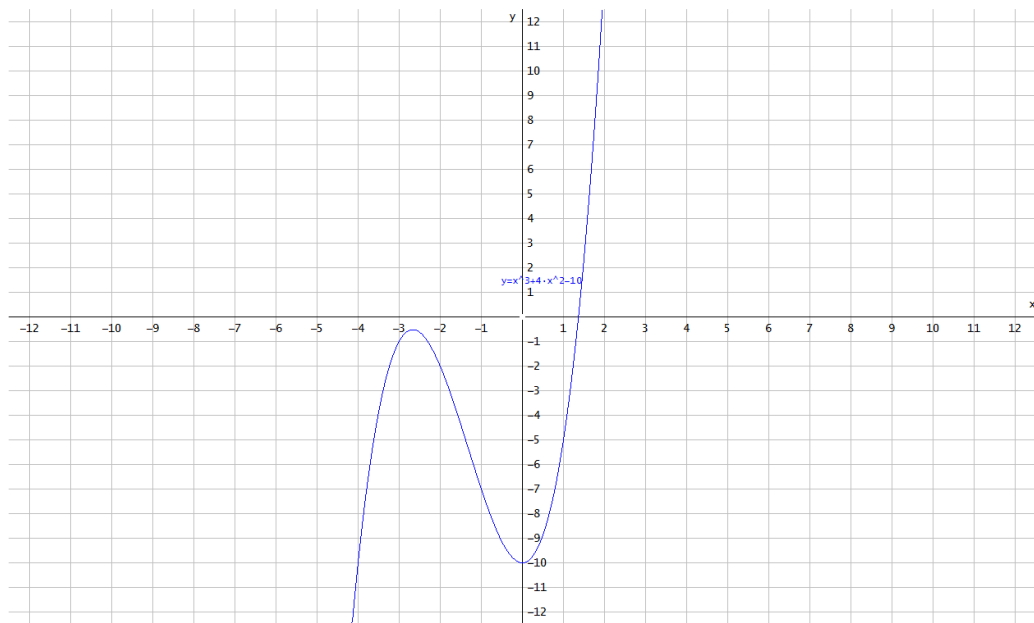
Paso 6: Si $f(a) \cdot f(x_n) < 0$, entonces tomar $b = x_n$, de lo contrario $a = x_n$

Paso 7: **Salida:** Se alcanzó N pero no la tolerancia.

- **Ventajas y desventajas de los métodos cerrados**

Para la ecuación $f(x) = x^3 + 4x^2 - 10$, se requiere su raíz con una tolerancia $\varepsilon = 10^{-3}$, la cual gráficamente se encuentra en el intervalo $[1,2]$ como lo muestra la figura.

Imagen 1. Gráfica de la función $f(x) = x^3 + 4x^2 - 10$



Se utilizarán ambos métodos cerrados con un máximo de 10 repeticiones para probar su efectividad y poder compararlos.

▪ **Solución usando el método de Bisección**

Para solucionar una ecuación algebraica no lineal manualmente utilizando cualquiera de los métodos que en este compendio se ilustren, siempre es bueno consignar los resultados en una matriz.

Siendo la ecuación para obtener el punto medio $x_n = \frac{a+b}{2}$ y aplicando el método se consiguen los siguientes resultados:

Tabla 4. Tabla de iteraciones del método de Bisección.

n	a	b	x_n	$f(a) \cdot f(x_n)$	$e_{R_n} = \left \frac{x_n - x_{n-1}}{x_n} \right $
1	1	2	1,5	-11,875	1
2	1	1,5	1,25	8,984	0,2
3	1,25	1,5	1,375	-0,2913	0,09090909
4	1,25	1,375	1,3125	1,5245	0,04761905
5	1,3125	1,375	1,34375	0,2978	0,02325581
6	1,34375	1,375	1,359375	0,0338	0,01149425
7	1,359375	1,375	1,3671875	-0,0031	0,00571429
8	1,359375	1,3671875	1,36328125	0,0031	0,00286533
9	1,36328125	1,3671875	1,365234375	-0,000002	0,00143062
10	1,36328125	1,365234375	1,3642578125	0,00052	0,00071582

En este método sólo se alcanza la tolerancia indicada en la última iteración, por lo cual se obtiene una raíz $\alpha^* = 1.3642578125$. A partir de la séptima iteración, la sucesión converge a 1.36.

▪ **Solución usando el método de falsa posición.**

Utilizando la ecuación $x_n = \frac{a_n \cdot f(b_n) - b_n \cdot f(a_n)}{f(b_n) - f(a_n)}$, para determinar el nuevo intervalo

en el que se encuentra acotada la raíz, se encontraron los siguientes resultados:

Tabla 5. Tabla de iteraciones del método de Falsa Posición.

n	a	b	x_n	$f(a) \cdot f(x_n)$	$e_{R_n} = \left \frac{x_n - x_{n-1}}{x_n} \right $
1	1	2	1,26315789	8,01137192	1
2	1,26315789	2	1,33882784	0,68956241	0,056519548
3	1,33882784	2	1,35854634	0,0473439	0,014514413
4	1,35854634	2	1,36354744	0,00305407	0,003667711
5	1,36354744	2	1,36480703	0,00019387	0,000922908
6	1,36480703	2	1,36512372	1,2257E-05	0,000231983
7	1,36512372	2	1,3652033	7,7416E-07	5,82959E-05
8	1,3652033	2	1,3652233	4,8882E-08	1,46484E-05
9	1,3652233	2	1,36522833	3,0863E-09	3,68073E-06
10	1,36522833	2	1,36522959	1,9486E-10	9,24862E-07

Usando Regula Falsi, la tolerancia es alcanzada a partir de la quinta iteración y la convergencia es visible a partir de la cuarta tendiendo el resultado a 1.365. En la décima iteración se obtiene una raíz $\alpha^* = 1,36522959$.

Para este ejemplo resulta más efectiva la utilización del método de Regula Falsi. De hecho en la mayoría de las veces, es más efectivo que el de bisección. Sin embargo, existen casos opuestos.

El método de bisección no tiene en cuenta la magnitud de los valores sino su signo, lo que hace que una aproximación intermedia e incluso mejor que la respuesta final pase desapercibida.⁶⁵ Por otro lado, presenta una convergencia muy lenta.

El método de Regula Falsi es mucho más efectivo en casos de convergencia y tolerancia. No obstante, la longitud del sub-intervalo que contiene a la raíz no tiende a cero por la concavidad de la gráfica de la función en las proximidades de la raíz, lo que hace que sea un extremo del intervalo el que cambie mientras que el otro permanece fijo a modo de pivote⁶⁶.

5.4.8. MÉTODOS ABIERTOS

Estos métodos no precisan de un intervalo para encontrar la raíz, de ahí su nombre. Sólo es necesario un valor inicial o una aproximación de la raíz para que el método pueda hallar el cero de la ecuación. Al no encerrar la raíz, al contrario de los métodos cerrados puede que las soluciones encontradas por éstos métodos

⁶⁵ Ibid, Pág 41.

⁶⁶ Ibid, Pág 42.

sean divergentes o se alejen de la raíz; sin embargo son más propensos a encontrar de manera más eficaz la solución de la ecuación.

- **Método de punto fijo**

Se dice que un número p es un punto fijo de una función cuando $g(p) = p$.

Si se parte del principio para encontrar una raíz de una ecuación que $f(x) = 0$, entonces $f(x)$ puede descomponerse en una función equivalente de x así,

$$f(x) = 0; f(x) = x - g(x) \Rightarrow g(x) = x$$

El método es convergente si y sólo si $|g'(x)| < 1$ para x en un intervalo $[a, b]$ que contiene a la raíz donde $g(x)$ es continua y diferenciable⁶⁷. Tal convergencia es 'más rápida' si $|g'(x)|$ es más pequeña.

El método de punto fijo para encontrar una raíz de α de la ecuación $x = g(x)$, consiste en generar la sucesión $\{x_n\}_n$ mediante la fórmula de iteración

$$x_n = g(x_{n-1}), n = 1, 2, 3, \dots, \text{ con } x_0 \text{ dado.}$$

El algoritmo para el método de punto fijo es el siguiente:⁶⁸

⁶⁷ Método de Iteración del Punto Fijo - Versión Electrónica <En Línea> URL: <http://docentes.uacj.mx/gtapia/AN/Unidad2/iteracion.htm>

Entrada: $g(x)$, Una aproximación inicial x_0 , una tolerancia ε , y un máximo de iteraciones N .

Salida: Un punto fijo aproximado α^* o un mensaje.

Paso 1: Tomar $n = 1$

Paso 2: Mientras que $n \leq N$ seguir los pasos 3-6:

Paso 3: Tomar $x_n = g(x_0)$

Paso 4: Si $|x_n - x_0| < \varepsilon$ entonces **Salida:** $\alpha^* = x_n$. **Terminar.**

Paso 5: Tomar $n = n + 1$

Paso 6: Tomar $x_0 = x_n$

Paso 7: **Salida:** Se alcanzó N pero no la tolerancia.

- **Método de Newton-Raphson**

La hipótesis general para aplicar el método de Newton-Raphson para hallar una raíz $\alpha \in (a, b)$ de una ecuación $f(x) = 0$, es que la función f tenga sus dos primeras derivadas continuas en el intervalo $[a, b]$ y $f'(x) \neq 0$, para todo $x \in [a, b]$.

En consecuencia, puede deducirse, que la función $f(x)$ tiene una pendiente definida y una única tangente en aquel intervalo. Así pues, por cada iteración del método, el punto donde la recta tangente a la curva de la función se hace igual a cero, se aproximará a la raíz de la misma.

⁶⁸ Asmar Ch., Op. Cit., Pág 46.

De esta manera, se puede iniciar la iteración desde la permisa de que la derivada en el punto inicial es igual a la ecuación de la recta tangente entre ese punto y el que toca a la curva, así:

$$f'(x_0) = \frac{f(x_0) - 0}{x_0 - x_1}$$

De tal forma que el nuevo valor de x , aproximado a la raíz de la ecuación será igual, por cada repetición a:

$$x_n = x_{n-1} - \frac{f(x_{n-1})}{f'(x_{n-1})}, \text{ donde } n \geq 1$$

Se puede hablar de este método numérico como una especialización del método de punto fijo, tomando como función de iteración a

$$g(x) = x - \frac{f(x)}{f'(x)}$$

La escogencia del punto inicial para empezar a iterar es muy importante, pues dependiendo de la cercanía del mismo a la raíz se puede llegar a ella con una mayor rapidez de convergencia.

Los criterios de aproximación que generalmente se utilizan en este método son: dado $\varepsilon > 0$, se toma como aproximación de la raíz α de la ecuación $f(x)=0$, al término x_N de la sucesión generada mediante la fórmula de iteración de Newton-Raphson, donde N es el menor entero no negativo tal que $|f(x_N)| < \varepsilon$ o $|x_N - x_{N-1}| < \varepsilon$. Si el método converge tal que,

$$|x_n - x_{n-1}| = \left| \frac{f(x_{n-1})}{f'(x_{n-1})} \right|$$

Entonces entre más grande sea $f'(x_{n-1})$ en la vecindad de la raíz α , más rápida será la convergencia.

El algoritmo para obtener una aproximación de una raíz de una ecuación $f(x)=0$ es el siguiente:

Entrada: $f(x)$, $f'(x)$, una aproximación inicial x_0 , una tolerancia ε , y un máximo de iteraciones N .

Salida: Una raíz aproximada α^* o un mensaje.

Paso 1: Tomar $n=1$

Paso 2: Mientras que $n \leq N$ seguir los pasos 3-8:

Paso 3: Tomar $e = f(x_0)$ y $d = f'(x_0)$

Paso 4: Si $d = 0 \Rightarrow$ **Salida:** No se puede continuar el método. **Terminar.**

Paso 5: Tomar $c = x_0 - \frac{e}{d}$, (Calcula x_n)

Paso 6: Si $|f(c)| < \varepsilon$ o $|c - x_0| = \left| \frac{e}{d} \right| < \varepsilon \Rightarrow$ **Salida:** Una raíz aproximada es $\alpha^* = c$. **Terminar.**

Paso 7: Tomar $n = n + 1$

Paso 8: Tomar $x_0 = c$, (Redefine x_0)

Paso 9: **Salida:** Se alcanzó N pero no la tolerancia.

- **Método de la secante**

Este método es similar al método de Newton-Raphson, sin embargo este obvia la inclusión de la derivada de la función al asumir que $x = x_{n-2}$ y

$$f'(x_{n-1}) = \lim_{x \rightarrow x_{n-1}} \frac{f(x) - f(x_{n-1})}{x - x_{n-1}}$$

Por tanto se aproxima $f'(x_{n-1}) = \frac{f(x) - f(x_{n-1})}{x - x_{n-1}}$, que no es más que la ecuación de

la pendiente de la recta secante a la gráfica de la función $f(x)$ por los puntos $(x_{n-1}, f(x_{n-1})), (x_{n-2}, f(x_{n-2}))$. Así pues, reemplazando $f'(x_{n-1})$ por su aproximación en la fórmula de iteración del método de Newton-Raphson, se obtiene la fórmula de iteración del método de la secante.

$$x_n = x_{n-1} - \frac{f(x_{n-1})(x_{n-1} - x_{n-2})}{f(x_{n-1}) - f(x_{n-2})}, n = 2, 3, \dots$$

Para implementar este método es imprescindible conocer dos puntos (aproximaciones) iniciales, x_0 y x_1 .

Existe una relación entre el método de la secante y el de Regula Falsi. Ambos buscan una aproximación a la raíz a partir de dos aproximaciones iniciales. La diferencia radica en que el método de la secante, los puntos iniciales no necesariamente deben encerrar a la raíz, lo que puede ocasionar divergencia para encontrar la raíz. Este método converge bajo los mismos criterios que el método de Newton-Raphson.

El algoritmo del método de la secante para encontrar una aproximación de la raíz de una función $f(x)$ entre dos aproximaciones iniciales es el siguiente:

Entrada: $f(x)$, dos aproximaciones iniciales x_0 y x_1 , una tolerancia ε , y un máximo de iteraciones N .

Salida: Una raíz aproximada α^* o un mensaje.

Paso 1: Tomar $n = 2, y_0 = f(x_0), y_1 = f(x_1)$

Paso 2: Mientras que $n \leq N$ seguir los pasos 3-7:

Paso 3: Si $y_1 - y_0 = 0 \Rightarrow$ **Salida:** No se puede aplicar el método. **Terminar.**

Paso 4: Tomar $x_2 = x_1 - \frac{y_1(x_1 - x_0)}{y_1 - y_0}$

Paso 5: Si $|x_2 - x_1| < \varepsilon \Rightarrow$ **Salida:** Una raíz aproximada es $\alpha^* = x_2$.

Terminar.

Paso 6: Tomar $n = n + 1$

Paso 7: Tomar $x_0 = x_1, y_0 = y_1, x_1 = x_2, y_1 = f(x_1)$

Paso 8: **Salida:** Se alcanzó N pero no la tolerancia.

- **Ventajas y desventajas de los métodos abiertos**

Se tomará la función $f(x) = x^3 + 4x^2 - 10$, se requiere su raíz con una tolerancia $\varepsilon = 10^{-3}$, la cual gráficamente se encuentra en el intervalo $[1,2]$ (Ver Imagen 1.

Gráfica de la función $f(x) = x^3 + 4x^2 - 10$ Imagen 1)

Se intentará encontrar la raíz de la función utilizando los métodos abiertos expuestos anteriormente para explicar sus ventajas y desventajas.

- **Solución usando el método de punto fijo.**

Para encontrar la raíz de una función mediante el método de punto fijo en un intervalo $[a,b]$ es necesario que el valor absoluto de la derivada de la función de punto fijo exista para todos los puntos del intervalo (a,b) y sea menor que uno.

Varias funciones de iteración pueden ser utilizadas, pero es necesario satisfacer el anterior teorema para seleccionar una de ellas.

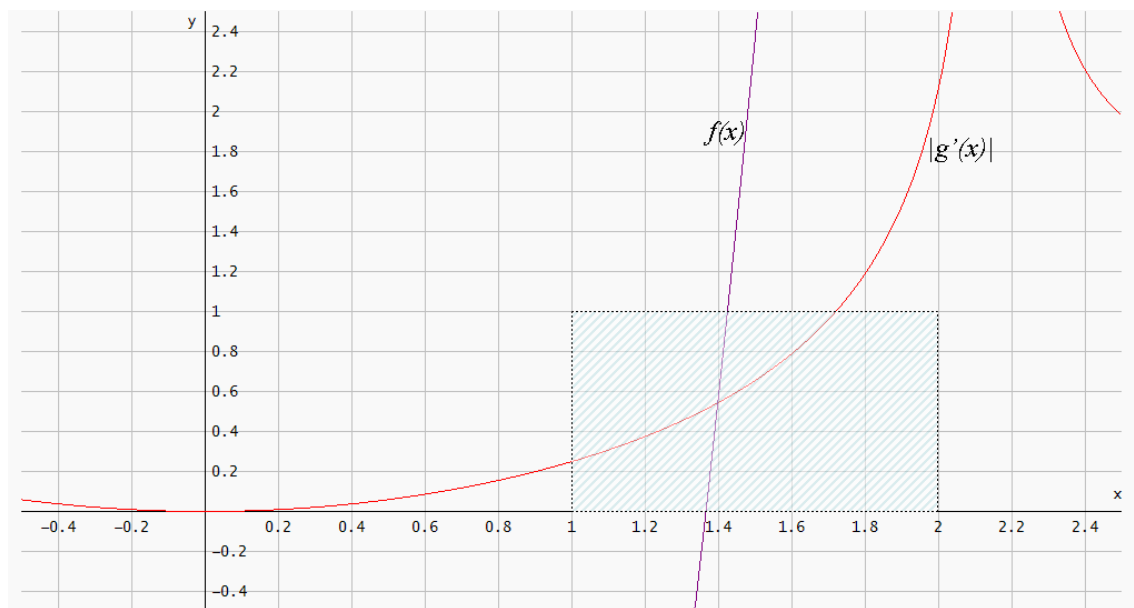
Se seleccionará la siguiente función para iterar.

$$x = g(x) \Rightarrow x = \sqrt{\frac{10 - x^3}{4}} \Rightarrow g(x) = \sqrt{\frac{10 - x^3}{4}}$$

$$g'(x) = -\frac{3x^2}{4\sqrt{10 - x^3}}$$

La función de iteración $g(x) = \sqrt{\frac{10-x^3}{4}}$, satisface el teorema de punto fijo, pues como se demuestra gráficamente, el valor absoluto de su derivada es menor que uno en la zona donde se encuentra la raíz de la función.

Imagen 2. Vecindad de la función $f(x) = x^3 + 4x^2 - 10$ en el intervalo $[1,2]$



Así pues, utilizando los siguientes parámetros iniciales, se prosigue a obtener la raíz de la ecuación:

$$x_{n+1} = \sqrt{\frac{10-x_n^3}{4}}; x_0 = 1$$

Tabla 6. Tabla de iteraciones del método del punto fijo

n	x_n	$e_{Rn} = \left \frac{x_n - x_{n-1}}{x_n} \right $	$< \varepsilon$
0	1		
1	1,5	0,5	FALSO
2	1,28695377	0,14203082	FALSO
3	1,4025408	0,08981444	FALSO
4	1,34545837	0,0406993	FALSO
5	1,37517025	0,02208309	FALSO
6	1,36009419	0,01096305	FALSO
7	1,36784697	0,00570017	FALSO
8	1,363887	0,00289503	FALSO
9	1,36591673	0,00148819	FALSO
10	1,36487822	0,00076031	VERDADERO
11	1,36541006	0,00038966	VERDADERO
12	1,36513782	0,00019938	VERDADERO
13	1,36527721	0,00010211	VERDADERO
14	1,36520585	5,2266E-05	VERDADERO
15	1,36524238	2,676E-05	VERDADERO

Realizando 15 iteraciones, el método alcanza la tolerancia a la altura de la décima iteración. Si se expone la rapidez de convergencia, la sucesión tiende notoriamente a la raíz $\alpha \approx 1.365230013$ a partir de la séptima iteración, dando saltos alrededor de un punto, el valor que se desea alcanzar, actuando como un pivote.

▪ **Solución usando el método de Newton-Raphson.**

El método de Newton-Raphson, al ser una especialización del método del punto fijo, cuenta con la ventaja inherente de poseer una función de iteración definida.

Así pues sólo es necesario conocer la derivada de la función para comenzar a aplicar ciclos sobre la función.

$$f'(x) = 3x^2 + 8x$$

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)} \Rightarrow x_{n+1} = x_n - \frac{x_n^3 + 4x_n^2 - 10}{3x_n^2 + 8x_n}$$

Aplicando el método de Newton-Raphson, es perceptible la vertiginosidad con la que se alcanzan tanto la tolerancia y la convergencia. La tolerancia se logra en la iteración 4 y en la iteración 6 es alcanzada la raíz de la función (Ver Tabla 7).

Se muestra una columna con el valor de la derivada de la función, pues es importante observar cuán grande se va haciendo este valor lo que va acelerando la convergencia (se duplica en cada iteración)

- **Solución usando el método de la secante.**

La función de iteración de este método pretende simular una recta secante a la función de tal forma que con cada ciclo, se vaya acercando a la raíz de la función. Esta función necesita dos valores iniciales que no serán necesariamente los extremos del intervalo, valores que se tomarán adrede para observar el comportamiento del método.

$$x_0 = 1; x_1 = 2.5; x_n = x_{n-1} - \frac{f(x_{n-1})(x_{n-1} - x_{n-2})}{f(x_{n-1}) - f(x_{n-2})}$$

Tabla 7. Tabla de iteraciones del método de Newton-Raphson.

n	x_n	$f'(x_n) = 3x_n^2 + 8x_n$	$e_{Rn} = \left \frac{x_n - x_{n-1}}{x_n} \right $	$< \varepsilon$
0	1			
1	1,45454545	11	0,45454545	FALSO
2	1,3689004	17,9834711	0,05888097	FALSO
3	1,3652366	16,5728681	0,00267646	FALSO
4	1,36523001	16,5135057	4,8246E-06	VERDADERO
5	1,36523001	16,5133991	1,558E-11	VERDADERO
6	1,36523001	16,5133991	0	VERDADERO
7	1,36523001	16,5133991	0	VERDADERO
8	1,36523001	16,5133991	0	VERDADERO
9	1,36523001	16,5133991	0	VERDADERO
10	1,36523001	16,5133991	0	VERDADERO
11	1,36523001	16,5133991	0	VERDADERO
12	1,36523001	16,5133991	0	VERDADERO
13	1,36523001	16,5133991	0	VERDADERO
14	1,36523001	16,5133991	0	VERDADERO
15	1,36523001		0	VERDADERO

El método de la secante alcanza la tolerancia en la sexta iteración y a partir de la misma se alcanza la raíz. Es convergente a partir de la quinta. A partir de la novena, no se puede seguir iterando pues la diferencia $f(x_{n-1}) - f(x_{n-2})$ es igual a cero por lo que el divisor es indefinido (Ver Tabla 8).

Tabla 8. Tabla de iteraciones del método de la secante.

n	x_n	$f(x_n)$	$f(x_{n-1}) - f(x_{n-2})$	$e_{Rn} = \left \frac{x_n - x_{n-1}}{x_n} \right $	$< \varepsilon$
0	1	-5			
1	2,5	30,625			
2	1,21052632	-2,36463041	35,625	0,515789474	FALSO
3	1,3029532	-0,9972451	32,9896304	0,076352642	FALSO
4	1,37036087	0,08494108	1,36738531	0,051734528	FALSO
5	1,36507002	-0,00264182	1,08218618	0,003860914	FALSO
6	1,36522961	-6,6337E-06	0,0875829	0,000116911	VERDADERO
7	1,36523001	5,2035E-10	0,00263519	2,94269E-07	VERDADERO
8	1,36523001	0	6,6342E-06	2,30809E-11	VERDADERO
9	1,36523001	0	5,2035E-10	0	VERDADERO

Queda demostrado, a pesar que los valores iniciales no son necesariamente los extremos del intervalo, al funcionar como recta secante a la función, siempre busca la raíz basándose en la función de iteración.

No obstante, esta ilustración, cada método puede ser eficaz o no según las circunstancias y, cada uno posee un grado de dificultad en la aplicación.

El método de punto fijo requiere se cumpla el teorema mencionado. (Véase Método de punto fijo) Por lo que deben probarse a menudo todas las alternativas posibles para que la función de iteración aplique.

El método de Newton-Raphson falla cuando un punto de inflexión ($f''(x) = 0$), en la vecindad de la raíz, porque itera sobre un máximo o mínimo de la función haciendo que la pendiente se acerque a cero y se aleje del área de interés. Tal desventaja también aplica al método de la Secante por la obtención de su función

de iteración obtenida a partir de la del método de Newton-Raphson. (Véase Método de la secante).

6. MARCO METODOLÓGICO

Para alcanzar las metas trazadas para esta investigación se han planeado una serie de actividades basadas en el proceso de la ingeniería de software, complementadas con el análisis sociológico que el proyecto requiere.

Se resaltan las siguientes actividades como las básicas dentro del proceso:

- Recopilación de información relacionada con sistemas tutores inteligentes, aplicables a métodos numéricos de ingeniería (fuentes bibliográficas, páginas Web, entrevistas).
- Análisis de herramientas metodológicas, sistemas tutores inteligentes existentes. Herramientas de desarrollo (diseño de interfaces, herramientas de análisis matemático, bases de datos, tomadores de decisión).
- Obtención de requerimientos del sistema (comprensión del dominio de aplicación, recolección y verificación de requerimientos).
- Abstracción de los requerimientos e identificación de componentes dentro del sistema (asignación de requerimientos, diseño de subsistemas, modelado en diagramas UML)
- Integración de componentes
- Pruebas del sistema
- Análisis del comportamiento del sistema con el usuario final.
- Entrega preliminar de informe final.

- Entrega definitiva del informe final y sustentación de la investigación.

7. DISEÑO DEL SISTEMA TUTOR INTELIGENTE

Siendo el objetivo de esta investigación, diseñar e implementar un sistema tutor inteligente basado en Web para la resolución de métodos numéricos de ingeniería aplicados a la solución de ecuaciones algebraicas no lineales, los autores, han procurado que el diseño, a pesar de ser básico, sea lo suficientemente abierto para que otras investigaciones, inclusive con un área de estudio diferente a la planteada en este compendio, puedan basarse en esta plataforma y adaptarla o mejorarla, conforme surjan los requerimientos de su contemporaneidad.

No obstante el diseño básico de un sistema tutor inteligente convencional, se componga de cuatro elementos primarios (modelo del tutor, modelo del alumno, modelo del dominio, interfaz), todos no serán objetos directos de este estudio. Una investigación paralela, tratará sobre la máquina de inferencia del sistema tutor, componente del modelo pedagógico, variante adoptado por los autores. Sin embargo, ambas, pretenden la consecución del mismo objetivo, por lo que el trabajo en equipo y el acoplamiento es preponderante para el éxito y la obtención del objetivo propuesto.

Se definen en este capítulo los aspectos preponderantes de diseño del sistema, de manera general. No obstante, una especificación minuciosa y los detalles de la implementación pueden consultarse posteriormente. (Ver Anexo A)

7.1. ESTRUCTURA DE OBTENCIÓN DEL CONOCIMIENTO

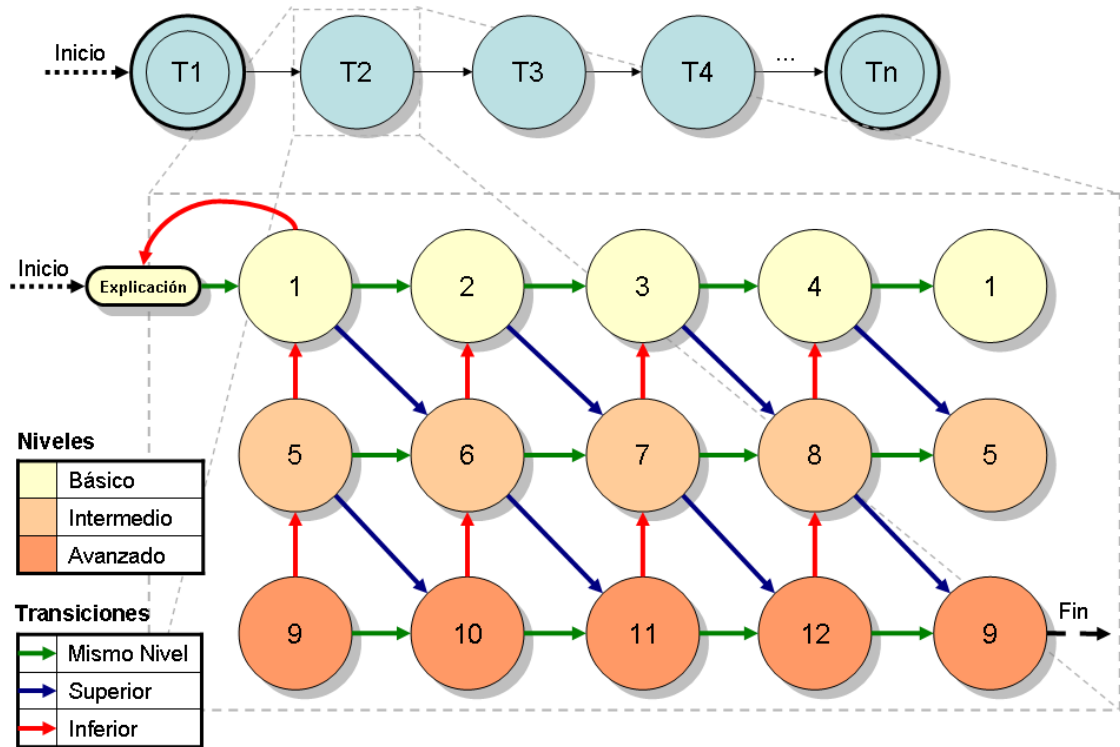
El conocimiento a impartirse durante una sesión debe ser alcanzado de manera progresiva y de tal forma, que le permita al estudiante retornar a cualquier punto estudiado con el objetivo de repasar o reforzar, o al sistema indicarle que temas debe volver a estudiar.

Un grafo dirigido es la alternativa ideal para representar este concepto. Cada nodo del grafo se traduce en un tema del contenido que pretende enseñarse.

Siendo i , la cantidad de temas que componen la estructura del conocimiento que se desea impartir y T un tema dentro de la misma, cada T_i es en sí mismo un grafo, compuesto por un conjunto de nodos, que representan problemas, clasificados en niveles de dificultad. (Ver Figura 7)

Un nodo es definido como un problema o un grupo de problemas que se le plantea al alumno, donde el sistema actúa basado en el principio socrático de la mayéutica, es decir, planteamiento de preguntas para comprobar en cada respuesta que el conocimiento enseñado previamente ha sido aprendido, y de ser así, ser promovido a un nodo de mayor dificultad o a un nuevo tema. En otras palabras, un nodo es una unidad de comprobación y análisis del conocimiento impartido.

Figura 7. Estructura de obtención de conocimiento.



Para fijar un nodo inicial para el proceso de aprendizaje, será necesaria una clasificación previa del estudiante. Por ello, es adecuado aplicar una prueba preliminar de contenidos básicos útiles para la aprehensión de los contenidos que a posteriori serán objeto de estudio. A partir de aquí, el sistema podrá seleccionar un nodo de mayor o menor dificultad según sean los resultados obtenidos en la misma.

Cada tema cuenta con niveles de dificultad. Por cada nivel, existe un rango de eficiencia permisible. De acuerdo a la eficiencia en la obtención de las respuestas

esperadas para el nodo actual y a su ubicación en el rango de permisividad por nivel, se elegirá un nodo siguiente así:

1. Máxima eficiencia. Eficiencia mayor que el límite superior del intervalo de permisividad del nivel de dificultad. Paso a un nodo de nivel superior.
2. Eficiencia regular. Eficiencia del nodo mayor o igual al límite inferior y menor o igual que el límite superior del rango permisible. Nodo del mismo nivel.
3. Eficiencia baja. Por debajo del límite inferior del intervalo. Nodo siguiente de nivel inferior.

Para calcular la eficiencia de un nodo, se tiene en cuenta que cada pregunta del mismo, tiene un porcentaje de ponderación. La sumatoria de los porcentajes de todas las preguntas que componen un nodo es igual al 100%.

Así pues, el cálculo de la eficiencia de un nodo se denota por la siguiente expresión:

$$E_N = \sum_{i=1}^n \frac{A_i}{R_i} \cdot P_i$$

Donde,

E_N , es la eficiencia del nodo.

n , es el número de preguntas que lo componen.

A , es el número de respuestas acertadas.

R , es el número de respuestas correctas posibles.

P , es el porcentaje ponderado por pregunta.

Cada respuesta dada, será almacenada en una base de datos del alumno. Con ello se pretende llevar un historial de obtención de conocimiento para cada uno de los estudiantes evaluados por el sistema. Esto es, guardar la respuesta dada y el número de intentos necesarios para alcanzar una respuesta esperada. Esto será útil, por ejemplo, si se han agotado todas las preguntas de un mismo nivel. El modo, representado en la eficiencia total de todos los nodos de un mismo nivel, cómo el estudiante se desempeñó en cada nodo, será el factor preponderante para seleccionar el nuevo nodo.

No obstante, cuando se pase a un nodo de un nivel visitado anteriormente y se obtenga una eficiencia menor al límite superior del intervalo de permisividad para nodos de ese nivel, existe una tolerancia por nivel que puede ser empleada sólo una vez por tema y nivel. Esto es, ampliar los límites del intervalo de permisividad por ambos lados el valor que la tolerancia indique.

En caso de que el alumno se encuentre en el nivel básico y obtenga una eficiencia baja, deberá ser remitido nuevamente a la explicación del tema para que repase los conocimientos enseñados. Si el alumno está en el último nivel que el tema ofrece, y obtiene la máxima eficiencia, podrá enseñársele el siguiente tema.

El sistema tendrá en cuenta para la promoción, mantención o degradación hacia un nodo siguiente el desempeño del alumno y las estadísticas de desempeño de nodos de igual nivel que el actual. Si el alumno se encuentra en un nodo de nivel básico, sólo podrá ir hacia la explicación del tema si su desempeño fue peor que el esperado. De igual manera, sólo podrá ir hacia delante si está en un nodo de

máximo nivel y logra una eficiencia superior a la esperada para el ejercicio. En un nodo de máximo nivel hacia atrás sólo irá a un nodo de nivel anterior.

Así, el alumno no podrá saltar entre temas si no ha superado la sesión de comprobación de conocimiento. Nunca podrá pasar de un nodo de nivel básico a un nodo de nivel alto inmediatamente y viceversa en un mismo tema.

El alumno podrá repasar todos los temas previos al tema actual, es decir, sólo podrá ver las explicaciones de cada tema. En ningún momento, deberá responder a la sesión de preguntas asociadas al tema si ya fue sometido a ella y la superó exitosamente.

Así pues, el conocimiento total es definido como un grafo dirigido donde la consecución del total de objetivos de un tema permitirá abordar el siguiente y no antes.

7.2. DISEÑO DE PREGUNTAS A FORMULAR

El área de estudio de la investigación, los métodos numéricos de ingeniería, requiere evaluar el análisis y no el resultado en sí. Esto es debido a la naturaleza de los problemas que pretenden resolverse empleándolos. Las soluciones que arroja la implementación de un método numérico son aproximaciones a las soluciones reales y basadas en coeficientes de error, por lo que es más factible

analizar el porqué de un resultado, no obstante, las limitaciones del algoritmo implementado o de las herramientas de cálculo utilizadas para ello.⁶⁹

Tomando como referencia observaciones sobre cuestionarios de evaluación formulados en libros de texto o exámenes generales, se pudieron inferir cinco tipos de preguntas básicas.

1. Completar un enunciado.
2. Falso – verdadero.
3. Selección múltiple con única respuesta.
4. Selección múltiple con múltiple respuesta.
5. Asociación.

Los autores de la investigación han modelado las preguntas como una correlación entre conjuntos de partida y de llegada, puesto que ésta es la característica que comparten los tipos de preguntas observadas.

Esto puede explicarse en que la selección que haga el estudiante, necesariamente parte de un conjunto de opciones, definidas implícita (completar un enunciado, falso o verdadero) o explícitamente, según sea el tipo de pregunta, y su respuesta es un subconjunto de las opciones presentadas. Vale aclarar, que el subconjunto que supone la respuesta no necesariamente es unitario, pues el número de elementos que lo componen lo determina el tipo de pregunta.

⁶⁹ *Ibíd.*, Pág 3.

La evaluación de la respuesta se simplifica comparando el conjunto de llegada del alumno con el que corresponde a la pregunta formulada. La inteligencia del tutor, se traduce en tomar una decisión a partir del acierto de la respuesta dada sobre lo correcto, es decir, del error que el estudiante haya podido tener.

Las preguntas cuyas opciones son definidas implícitamente, como pueden ser las de 'completar un enunciado' o de 'falso o verdadero', son clasificadas así, pues las opciones de respuesta son fáciles de intuir o sólo admiten una única opción de respuesta válida.

Las respuestas a las preguntas de tipo 'complete un enunciado', van atadas a la teoría explicada y no admiten ambigüedades, pues es sólo una frase, en este caso la faltante, la que da sentido completo a toda la expresión.

Imagen 3. Ejemplo de pregunta de tipo 'complete el enunciado'

Los métodos analíticos se fundamentan teóricamente en el Teorema de Bolzano. Completa su enunciado.

Si $f(x)$ es una función _____ en el intervalo $[a,b]$ y cambia de signo en los extremos de ese intervalo, entonces existe un número $c \in$ ____ tal que: $f(c) = 0$.

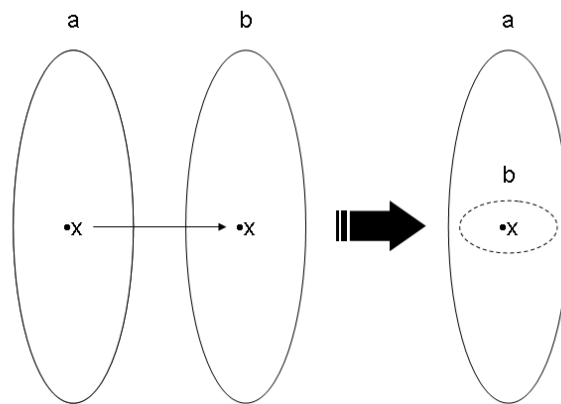
Si además f es estrictamente _____ entonces la raíz es única.

(En este caso las palabras subrayadas deben ser suministradas por el alumno).

De este modo, el conjunto de respuestas posibles (a) para una pregunta de este tipo, es unitario (conjunto de partida). Su conjunto de llegada (b), será un conjunto equivalente, debido a lo explicado previamente. (Ver Figura 8)

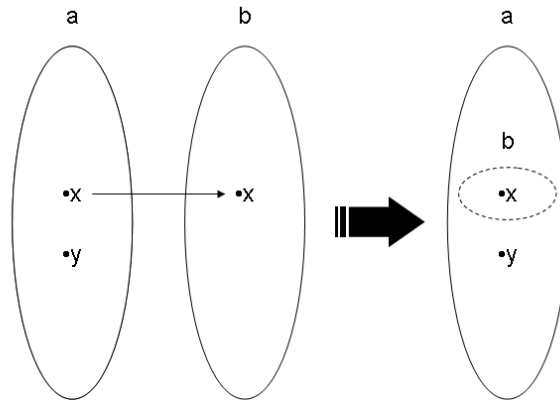
Para todos los tipos de preguntas, el conjunto de llegada es un subconjunto del conjunto de partida. La base para la evaluación de un alumno, es la intersección entre el conjunto de respuesta que haya dado y el conjunto de llegada que corresponde a las respuestas correctas.

Figura 8. Conjuntos de partida y llegada para una pregunta de tipo I.



De igual forma, las preguntas de tipo 'falso o verdadero', tampoco dan cabida a la ambigüedad, pues de los valores posibles sólo uno es susceptible de seleccionarse. (Ver Figura 9)

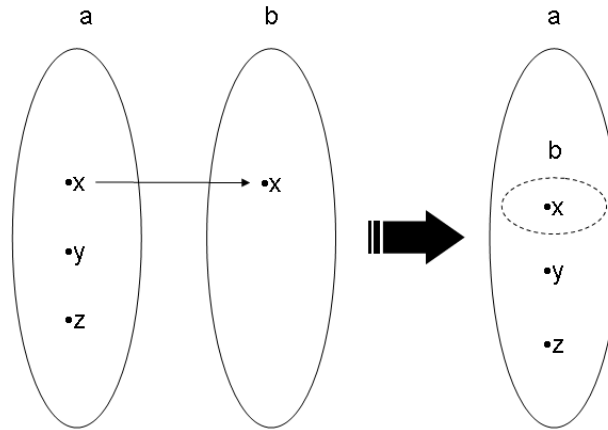
Figura 9. Conjuntos de partida y llegada para una pregunta de tipo II.



Las preguntas cuyas opciones de respuesta están definidas explícitamente, (III – Selección múltiple con única respuesta, IV – Selección múltiple con múltiple respuesta, V – Asociación), presentan un comportamiento similar, lo que facilita el modelamiento.

Una pregunta de selección múltiple con única respuesta se diferencia de una de tipo 'falso o verdadero' porque el número de opciones es mayor que dos y las opciones están definidas según sea el tema de estudio. De otro modo, el manejo es igual.

Figura 10. Conjuntos de partida y llegada para una pregunta de tipo III.



Para modelar una pregunta de tipo ‘selección múltiple con múltiple respuesta’, el conjunto de partida se traduce en las posibles combinaciones de las opciones que ofrece la pregunta.

Por ejemplo para una pregunta de cuatro opciones, el máximo conjunto de opciones que supondría una respuesta correcta estaría compuesto de cuatro opciones. El número de elementos descendería a un único ítem para la respuesta correcta.

Así pues siendo n , el número de opciones para la respuesta, el número de elementos del conjunto de partida para una pregunta de tipo IV, se define con la siguiente expresión:

$$\sum_{i=1}^n nC(n+1-i)$$

7.3. ARQUITECTURA DEL SISTEMA TUTOR

7.3.1. ASPECTOS DE DISEÑO

El sistema tutor inteligente que pretende implementarse a través del desarrollo de esta investigación, se basa en la arquitectura estándar de un STI convencional: modelo del dominio, modelo del tutor, modelo del alumno e interfaz. (Ver Figura 2). No obstante, estos cuatro módulos, además de ser el fundamento de cualquier investigación afín a los sistemas tutores inteligentes, los autores de este proyecto procuraron tener en cuenta los siguientes aspectos de diseño o requerimientos no funcionales.

- **Mantenibilidad:** Revisión y renovación de los contenidos, estructuras visuales y de datos según los requerimientos del tiempo.
- **Confiability:** No deberá causar perjuicios al entorno en el que pretende ser utilizado.
- **Interactividad:** El sistema deberá permitir la interactividad necesaria con el estudiante para mantener su atención y permitir el aprendizaje.
- **Adaptabilidad de contenidos:** El sistema deberá ser lo suficientemente abierto, en sus estructuras funcionales (estructuras de datos, reglas de inferencia, comunicación entre componentes), para posibilitar su uso en contenidos de aprendizaje distintos a los planteados en este compendio.

- Seguridad: El sistema deberá segregar entre usuarios e identificarlos por un nombre de usuario y contraseña. Ningún usuario podrá acceder al historial de aprendizaje de otro.
- Integridad de datos: La inserción, actualización y eliminación de información deberán preservar la coherencia entre las estructuras de datos que la almacenan.

Respecto a la arquitectura del sistema en sí, los autores de este proyecto adoptan la variante propuesta por Huapaya⁷⁰, que se traduce en la separación del componente pedagógico del modelo del tutor. (Ver Figura 12) La admisión de esta idea se explicará más adelante.

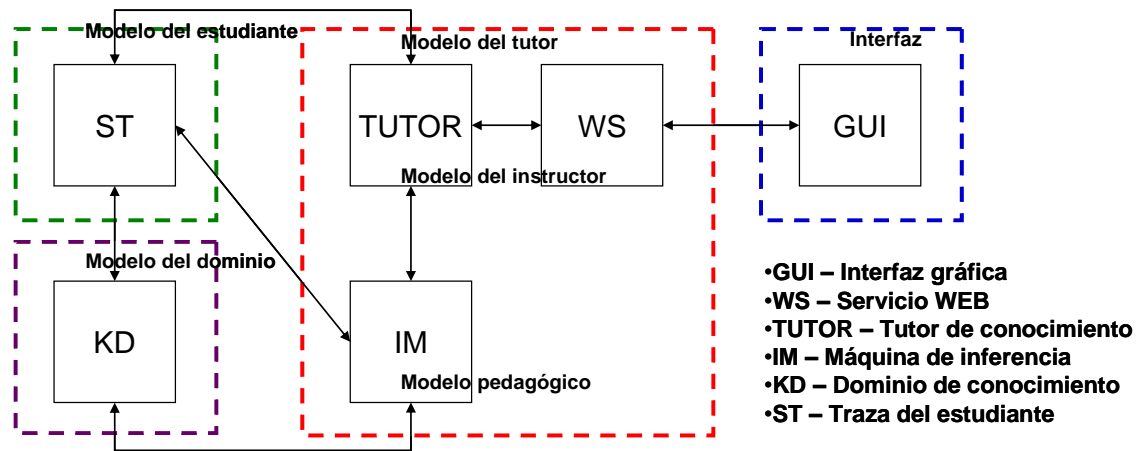
Para posibilitar su uso en Web, el sistema incluye en su arquitectura un Web Service, puerta de enlace entre los componentes básicos y la interfaz de usuario, para atender las peticiones de varios usuarios a través de la red, procurando la enseñanza desde cualquier lugar. El postulado base para el desarrollo de esta idea fue propuesto por Alpert en un sistema tutor dedicado a estudiantes de álgebra.⁷¹

Cada uno de los modelos que lo integra, se ve traducido en los siguientes componentes de software.

⁷⁰ Huapaya, C. R., Arona, G. M. y Lizarralde, F. A. – Op. Cit. Pág. 3

⁷¹ Alpert, Sherman, Singley, Mark y Fairweather, Peter – Deploying Intelligent Tutors on the Web – International Journal of Artificial Intelligence in Education, 1999 – Versión Electrónica <En Línea> URL: http://aied.inf.ed.ac.uk/abstracts/Vol_10/alpert.html

Figura 12. Arquitectura del STI implementado en la investigación.



7.3.2. INTERFAZ GRÁFICA (GUI)

La interfaz gráfica del STI, fruto de esta investigación, deberá ser dinámica, fácil de utilizar de tal manera, que al alumno le sea asequible navegar entre contenidos y pueda mantener su atención en lo que se le está enseñando. Debe ser fácil de mantener y ser actualizable conforme a los nuevos requerimientos de su tiempo.

La interfaz debe contar con agentes que sirvan de apoyo, a la enseñanza de los métodos numéricos en este caso, como animaciones, graficadores de funciones, simulación de ejercicios para posibilitar el didacticismo.

Sin embargo, también debe ser lo suficientemente abierta para poder adaptarse a cualquier otro tipo de contenido diferente al que aborda esta investigación,

inclusive sus agentes didácticos de soporte, o permitir el acoplamiento de agentes especializados en un área de conocimiento.

Se desea que el sistema tutor esté implementado en Web, por lo que la interfaz gráfica debe poder visualizarse o descargarse a través de cualquier computador que tenga acceso a Internet y a su vez debe permitir la comunicación remota con el sistema tutor, para facilitar el flujo de aprendizaje en una sesión de enseñanza.

7.3.3. SERVICIO WEB

Una de la bases de la implementación de esta investigación, es que el sistema tutor este soportado bajo una plataforma Web, que posibilite la enseñanza aún fuera del entorno físico de una universidad.

Los autores de este proyecto, han determinado que la manera más adecuada de implementar esta idea es mediante un servicio Web. Bajo protocolos simples de comunicación, la interfaz, módulo portable con el cual el alumno interactúa, se enlaza con el servicio Web para intercambiar mensajes entre el alumno y el tutor en una sesión de enseñanza.

El servicio Web provee independencia, en aras de lograr una aplicación durable y fácil de mantener en el tiempo, por lo que definiendo una política de comunicación estándar entre el y la interfaz, se pueden construir interfaces a la medida según los requerimientos de lo que se quiera enseñar.

El fundamento de su implementación radica en separar del módulo tutor, las funciones que permiten la comunicación entre aquél y un estudiante. Inclusive, mejorar esta característica dándole la capacidad de aceptar y procesar más de una petición, ampliando la cobertura de enseñanza a muchos estudiantes obviando la concurrencia en el tiempo.

7.3.4. TUTOR

El módulo tutor, debe servir de puente entre el alumno, a través de la interfaz gráfica, y la máquina de inferencia. Por ello debe ser capaz de establecer comunicación con la máquina de inferencia, activar el servicio Web para la admisión de peticiones (concurrencia de alumnos) y la interfaz gráfica de modo que pueda transmitir los mensajes entre uno y otro (explicaciones, preguntas, respuestas y retroalimentación de las respuestas), con el objetivo que cada receptor comprenda la información y pueda responder a ella. El módulo tutor es responsable de registrar el guarismo de desempeño del estudiante por cada nodo visitado en la base de datos del alumno. Es decir, es el responsable de construir la traza de seguimiento del alumno. Por lo que deberá establecer comunicación con aquella, registrar la información según lo requieran sus estructuras de datos y construir la traza de seguimiento del alumno de manera coherente.

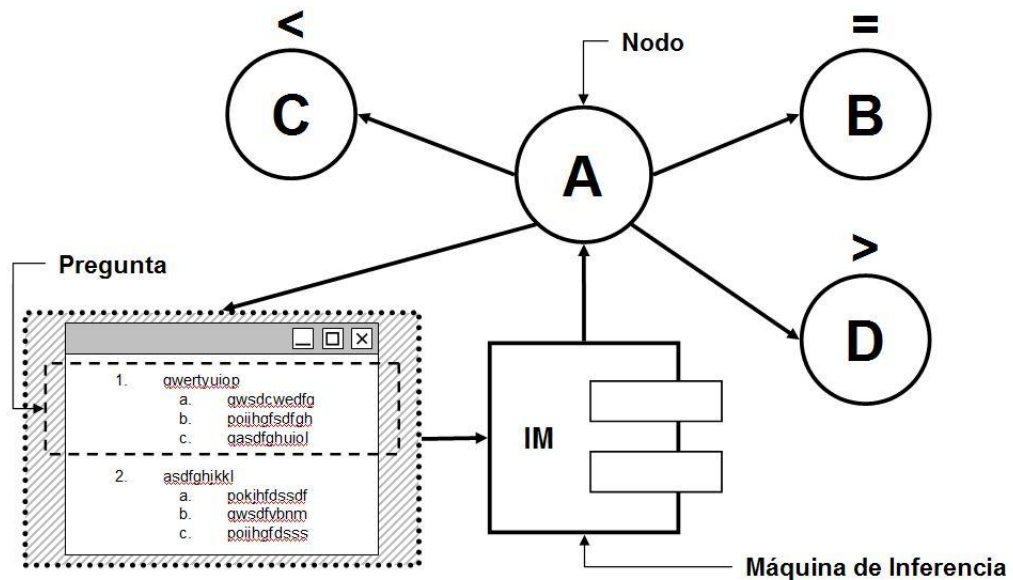
7.3.5. MÁQUINA DE INFERENCIA.

La máquina de inferencia del sistema tutor que quiere implementarse por medio de esta investigación, debe ser capaz de entablar comunicación con el módulo tutor y realizar las respectivas inferencias a cada respuesta que dé el alumno, basándose no solo en la respuesta en sí, sino también del comportamiento del alumno (traza de seguimiento), durante todo el aprendizaje (todas las sesiones que haya realizado el alumno para llegar al punto actual).

La razón de adoptar el postulado presentado por Huapaya, es porque a través de la máquina de inferencia se construye la metodología a seguir, de acuerdo al mismo comportamiento del alumno. Así, si un alumno demuestra a través de las respuestas que no ha alcanzado el nivel de aprehensión requerido para promoverlo al siguiente, el sistema tutor a través de la máquina de inferencia basado en su traza, podrá disminuir los niveles de dificultad ofreciéndole al alumno facilidades para la comprensión de los contenidos, tales como problemas de menor dificultad o, adaptando la explicación del tema o la manera como se ilustra. Siendo un nodo un cúmulo de preguntas, la máquina de inferencia para obtener la eficiencia del nodo deberá tener en cuenta el desempeño en términos de respuestas acertadas y la ponderación de cada pregunta. Así, y conociendo el rango de permisividad por el nivel de dificultad del nodo, tomará la decisión de promover, degradar o mantener al estudiante en un nivel de dificultad en el

siguiente nodo. (Ver Figura 13). Este componente será diseñado e implementado por una investigación paralela.

Figura 13. Selección de un nodo siguiente.



7.3.6. BASE DE DATOS DEL ALUMNO

El almacén de datos del alumno, se construye a partir de la información personal y académica inclusive, de aquel. Además, recopilará su historial de aprendizaje. Esto no es más que los nodos visitados uno a uno para la consecución del conocimiento. Este historial deberá almacenar la progresión alcanzada en cada nodo, es decir, respuestas, número de intentos necesarios para alcanzar la respuesta correcta. Para poder construir el camino recorrido por el alumno, deberá

almacenarse el nodo anterior visitado y el siguiente nodo que visitará, según sea su desempeño. El responsable de realizar el almacenaje será el módulo tutor.

Los datos almacenados de cada alumno, serán útiles para que la máquina de inferencia pueda deducir a partir del desempeño del estudiante, cuál puede ser el siguiente ejercicio que podrá plantear al alumno.

7.3.7. BASE DE DATOS DE CONOCIMIENTO

La base de datos de conocimiento, almacena los contenidos a enseñar, para que el módulo tutor los muestre en las explicaciones y los problemas, y sus respuestas, a plantear por parte de la máquina de inferencia.

Todos los nodos deben estar estructurados según el plan de obtención de conocimiento, explicado anteriormente, buscando la sencillez al momento de construir completamente la traza de seguimiento del alumno.

Cada nodo que se almacene debe indicar a qué tema pertenece, su nivel de dificultad y a qué nodo debe apuntar, según lo requiera el desempeño y las circunstancias de aprendizaje.

7.4. DISEÑO DE ESTRUCTURAS DE DATOS

En consecuencia con lo planteado, respecto al modelado del conocimiento, el sistema tutor inteligente, el diseño de las estructuras de datos que almacenarán la información circundante entre sus componentes, se fundamenta en el seguimiento de la obtención, por parte del alumno, de los saberes impartidos a través del sistema y en los nodos que componen la estructura total de conocimiento, como las unidades de comprobación del mismo.

La integridad de la información, como aspecto de diseño, se cimienta en la coherencia de la misma, almacenada en cada entidad de datos y en las relaciones entre ellas.

Se definen entonces, estructuras de datos para definir los siguientes aspectos:

- Identificación unívoca de cada usuario del sistema y control de acceso: Cada usuario está registrado con un nombre y contraseña (login) para ingreso a una sesión de enseñanza – aprendizaje y un número único con el cual asociar su traza de seguimiento. El sistema también almacena la información personal del estudiante, la cual asocia con el número único asignado.
- Modelado del conocimiento: Las unidades de comprobación de conocimiento, o nodos, se almacenan con su respectiva respuesta y retroalimentaciones. Cada nodo hace parte de un tema. Los temas se modelan como estructuras de alto nivel dentro de la jerarquización del

conocimiento. Cada nodo está clasificado en niveles de dificultad. El nivel es una propiedad inherente de cada nodo, por lo que se modela dentro de la estructura lógica del mismo. Cada nodo debe almacenar una eficiencia mínima requerida para comparar los resultados del alumno con los esperados. Los nodos están identificados con números únicos asignados por el sistema al momento de ser creados. Las entidades que se relacionan con los nodos lo hacen a través de esta tipificación.

- Consecución del conocimiento: Para la construcción del recorrido de conocimiento realizado por el alumno, se almacena por cada uno, cada nodo visitado y su desempeño traducido en número de intentos realizados por nodo, el nodo anterior visitado y el nodo siguiente según sea la decisión de la máquina de inferencia.

8. CONCLUSIONES

La utilización de un sistema tutor inteligente dentro de los contenidos de estudio de un área de conocimiento en particular, no pretende la suplantación de un docente presencial, ni mucho menos humano, pues es bien sabido que la inteligencia artificial es un área en desarrollo. Más bien, un STI es una herramienta auxiliar que siendo didáctica, en la presentación de contenidos y ayudas, es inteligente pues puede inferir sobre los resultados obtenidos en una prueba planteada y determinar el nivel de conocimiento del estudiante con base en ello, tal como lo hiciera un tutor humano. Pero, más allá de esta ventaja que supondría sobre un tutor, puede emplearse como un complemento a la enseñanza y a la preparación de un curso, de métodos numéricos para el caso particular de esta investigación.

La representación de la obtención del conocimiento en nodos o unidades en grafos dirigidos, otorga la ventaja de construir un historial de seguimiento. Pero esto es debido a definir como propiedades o características inherentes de cada nodo un tema asociado y un nivel de dificultad. La consecución de nodos dentro de un tema constituye sub-grafos dentro del grafo base que supone la colección de todos los temas o tópicos de enseñanza.

Así, decisión a decisión, fundamentado en cada respuesta que el alumno dé a una pregunta formulada y en las propiedades ya mencionadas, se va construyendo estas estructuras lógicas, que son diferentes por cada alumno.

Los autores quieren dejar abierta la posibilidad de variar la amplitud de la estructura de conocimiento. Es decir, la consumación de cada tema puede alcanzarse al superar N niveles de dificultad y/o cada nivel de dificultad puede contar con una cantidad indeterminada de nodos; esto según sean los requerimientos de cada área de conocimiento. Esto es irrelevante comparado con la distinción unívoca del nivel de dificultad de cada nodo, para facultar a la máquina de inferencia, componente del sistema encargado de la toma de decisiones, la selección de un nodo siguiente según el desempeño o eficiencia del alumno en la resolución de un problema planteado.

La asignación de roles a cada componente del sistema, es muy importante para el cumplimiento del objetivo de enseñanza. No obstante exista un prototipo de arquitectura básica para el diseño de sistemas tutores inteligentes, implementar según las nociones de aquella o, para el caso de este estudio, una variación de la misma es tarea de cuidado.

El análisis de la labor de cada componente debe realizarse, procurando preservar tales aspectos, inclusive en aras de cumplir los objetivos propuestos para la investigación.

Los autores han seguido dos variaciones propuestas por otras investigaciones, dentro de sus lineamientos:

- Separación del componente tomador de decisiones del módulo tutor al módulo pedagógico: Propuesta por Constanza Huapaya (Ver ARQUITECTURA DEL SISTEMA TUTOR). La modularización de esta manera, permitió en primera instancia, la distribución del diseño e implementación del sistema tutor en dos investigaciones, la descrita en este estudio y una paralela sobre la máquina de inferencia y el uso de agentes expertos. La implementación de una en la otra, para el acoplamiento descrito en la arquitectura (Ver Figura 12), se logró gracias al paradigma de la programación orientada a objetos: La máquina de inferencia es un objeto asociado al componente tutor, lo que genera independencia de implementación pero ocasiona a su vez alta cohesión entre ambos componentes. Ambos componentes son indefectiblemente copartícipes. La máquina de inferencia es uno de los principales actores dentro del proceso. Sin embargo, el principal actor es el alumno mismo, pues con su desempeño condiciona su camino dentro del proceso de enseñanza.
- Segregación del componente tutor en cliente y servidor para ofrecer un STI basado en Web: Propuesta por Sherman Alpert.⁷² El alumno es libre de aprender en cualquier momento y en cualquier lugar. La Web ofrece la ventaja de llevar la enseñanza a más usuarios, por su amplio alcance. Bajo este precepto, y liberando al usuario de una instalación 'standalone' en un equipo de cómputo, se desmiembra del componente tutor, un nuevo

⁷² *Ibíd.*

componente, el Web Service implementado para atender las solicitudes de usuarios simultáneamente.

Basados en la adaptabilidad o la apertura a cualquier tipo de contenido, un objetivo de diseño y una característica inherente de un STI, los autores han pretendido sentar las bases de un STI pionero y plataforma de apoyo. Reingeniería y reutilización de los recursos empleados (arquitectura del sistema, estrategias tutoriales, contenidos), son la puerta para la realización de este postulado.

El STI empleó como herramienta de enseñanza la adaptabilidad en comprobación de enseñanza. Esto es, aumento o disminución del nivel de dificultad en las preguntas, según el desempeño del alumno.

Al ser un proyecto básico, inclusive por circunstancias de alcance y tiempo, los autores proponen escribir nuevos incrementos para el sistema, continuando en el fundamento de la adaptabilidad, pero expandiéndolo de acuerdo a las siguientes estrategias como líneas futuras de investigación o puntos de partida para la reingeniería:

- Adaptabilidad en contenidos de enseñanza: Diseño e implementación de un componente súper-tutor, que pueda ser capaz de generar nuevas explicaciones en lenguaje natural, distintas a las explicaciones básicas, en consecuencia de los requerimientos del estudiante (explicaciones necesarias para bajos niveles de aprehensión o nuevas explicaciones solicitadas por el propio alumno). Además, pueda ser capaz de lanzar sus propias preguntas o ejercicios (creación de nodos sin intervención humana),

a través del auto-aprendizaje basado en técnicas de la inteligencia artificial. Esto, facilitando la labor de la máquina de inferencia en la selección de nodos cuando ya se hayan agotado todos los predefinidos para un mismo nivel de dificultad.

- Adaptabilidad en explicación de contenidos: Esta línea de investigación sugiere el diseño e implementación de un componente capaz de establecer y comprender un diálogo en lenguaje natural con el alumno en aras de resolver sus inquietudes a lo largo de una sesión de enseñanza. Puede manifestarse en el desarrollo de un agente animado que acompañe el proceso de aprendizaje dando soporte y ayudas al alumno cuando este lo requiera o dando sugerencias cuando aquél presente dificultades en la comprensión de contenidos impartidos. Estos dos tópicos de adaptabilidad se fundamentan en la contextualización del sistema al alumno; es decir, modelar el sistema, su base teórica (adaptabilidad de contenidos) y/o su base metodológica (adaptabilidad en explicación) al nivel de aprehensión que demuestre el estudiante durante el proceso cognitivo.
- Adaptabilidad en entornos visuales: Diseño e implementación de un componente, que permita la reacomodación de los componentes hipertexto (texto, audio, imágenes, vídeo), según las preferencias o necesidades de aprendizaje del alumno.

FUENTES DE REFERENCIA

- Alpert, Sherman, Singley, Mark y Fairweather, Peter. Deploying Intelligent Tutors on the Web. International Journal of Artificial Intelligence in Education, 1999 – Versión Electrónica <En Línea> 1999. URL: http://aied.inf.ed.ac.uk/abstracts/Vol_10/alpert.html
- Anderson, J. R. The expert module. Carnegie-Mellon University. 1988 – Versión Electrónica <En Línea> 2003. URL: <http://act-r.psy.cmu.edu/papers/78/Expert.Module.pdf>
- Applications of AI in Education – Crossroads: The ACM Student Magazine – Versión Electrónica <En Línea> 2007. URL: <http://www.acm.org/crossroads/xrds3-1/aied.html>
- Arenas, C. Fichas Matemáticas. – Versión Electrónica <En Línea> 2007. URL: <http://diposit.ub.edu/dspace/bitstream/2445/1322/1/218.pdf>
- Asmar Ch., Iván – Métodos Numéricos. Un primer curso – Medellín: Universidad Nacional de Colombia Sede Medellín, Facultad de Ciencias, Departamento de Matemáticas, 1999.
- Botti, Julián. Agentes Inteligentes: el siguiente paso en la inteligencia artificial. Versión Electrónica <En Línea> 2000. URL: <http://www.ati.es/novatica/2000/145/vjulia-145.pdf>

- Brusilovsky, Peter. Adaptive hypermedia: an attempt to analyse and generalize – International Centre for Scientific and Technical Information. Versión Electrónica. <En Línea> 1999. URL: <http://wwwis.win.tue.nl/ah94/Brusilovsky.html>
- Castellano M., Hugo – Informática Educativa – Versión Electrónica <En Línea> 2004. URL: <http://cie.ilce.edu.mx/sitio/academica/Inform%E1tica%20educativa.pdf>
- Colabora.NET: Conceptos fundamentales de la POO (Programación orientada a objetos) – Versión Electrónica <En Línea> 2005. URL: http://www.elguille.info/colabora/puntoNET/canchala_FundamentosPOO.htm
- Cute Paste – Glosario. Versión Electrónica <En Línea> 2007. URL: <http://cutepaste.wordpress.com/glosario-y-definiciones/>
- De Llano, José Gregorio, Adrián, Mariela - La Informática Educativa en la Escuela. Fe y Alegría – Fundación Santamaría. Caracas, 1998.
- Del Valle Sierra, Jesús. Curso de Matemáticas dirigido a Estudiantes de Ingeniería de Sistemas. – Versión Electrónica <En Línea> 2001. URL: http://huitoto.udea.edu.co/Matematicas/Linea_Recta_Def.htm
- Elguea, Javier. Inteligencia artificial y psicología: la concepción contemporánea de la mente humana. Versión Electrónica. 2007. <En Línea> URL: http://biblioteca.itam.mx/estudios/estudio/estudio10/sec_16.html

- Fernández, Luisa y otros. Sistemas Hipermedia Adaptativos: una aproximación al tema. Revista Cubana de Informática Médica – CECAM. Versión Electrónica <En Línea> 2004. URL: http://www.cecam.sld.cu/pages/rcim/revista_5/articulos_hm/hipermedia.htm
- González, Carina S. - Sistemas Inteligentes en la Educación: Una revisión de las líneas de investigación y aplicaciones actuales - Versión Electrónica <En Línea> 2005. URL: http://www.uv.es/RELIEVE/v10n1/RELIEVEv10n1_1.pdf
- Hernández S., Elsie. Calculo Diferencial e Integral. – Versión Electrónica <En Línea> 2006. URL: <http://www.cidse.itcr.ac.cr/cursos-linea/CALCULODIFERENCIAL/curso-elsie/limitesycontinuidad/html/limitesfinal.pdf>
- Huapaya, C. R., Arona, G. M. y Lizarralde, F. A. - Enseñanza de la Ingeniería con Sistemas Tutoriales Inteligentes - Versión Electrónica <En Línea> 2005. URL: http://www.scielo.cl/scielo.php?pid=S0718-07642005000500012&script=sci_arttext&tlng=es
- Impacto de la informática en la educación. Educación – Ciberhábitat, Ciudad de la Informática – Versión Electrónica <En Línea> 2007. URL: <http://ciberhabitat.gob.mx/universidad/ui/eadei/eadeii.htm>
- Impacto de la informática en la educación. Informática educativa – Ciberhábitat, Ciudad de la Informática – Versión Electrónica <En Línea> 2007. URL: <http://ciberhabitat.gob.mx/universidad/ui/eadei/eadev.htm>

- Impacto de la informática en la educación. Tecnología – Ciberhábitat, Ciudad de la Informática – Versión Electrónica <En Línea> 2007. URL: <http://ciberhabitat.gob.mx/universidad/ui/eadei/eadeiv.htm>
- Inteligência Artificial na Educação – Versión Electrónica <En Línea> 2002. URL: <http://www.nce.ufrj.br/ginape/publicacoes/trabalhos/RenatoMaterial/iaeducacao.htm>
- Inteligencia Artificial. Versión Electrónica <En Línea> 2007. URL: http://www.inteligenciaartificial.cl/ciencia/software/ia/inteligencia_artificial.htm
- Jiménez Rey, Elizabeth, Grossi, Maria Delia y otros. Diseño de un Sistema Tutor Inteligente Multiagente. Revista Eletrônica de Sistemas de Informação, Edição 8, nº2. 2006.
- Leyva, Hugo Pablo - Métodos Numéricos. Teoría de Errores. <En Línea> 2003. URL: <http://sai.azc.uam.mx/apoyodidactico/mn/Unidad1/mntema1.html>
- Loncán, Pierre. Estructura de la Información. Versión Electrónica <En Línea> 2007. URL: http://curso.ihmc.us/servlet/SBReadResourceServlet?rid=1201385077671_1590506156_7659

- Medina, Nuria. Taxonomía de Sistemas Hipermedia Adaptativos. Universidad de Granada. Versión Electrónica. <En Línea> 2002. URL: <http://lsi.ugr.es/~fguti/taller/06/Nuria%20Medina.pdf>
- Método de Iteración del Punto Fijo - Versión Electrónica <En Línea> 2001. URL: <http://docentes.uacj.mx/gtapia/AN/Unidad2/iteracion.htm>
- Millán, Eva. Sistema Bayesiano para el Modelado del alumno – Universidad de Malaga. Tesis doctoral. Versión Electrónica. <En Línea> 2000. URL: www.lcc.uma.es/~eva/investigacion/SBMA.pdf
- Padilla Franco, Jávitt, Higmar, Nahitt y otros. Sistema Inteligente para la enseñanza de las Matemáticas. – Versión Electrónica <En Línea> 2001. URL: www.depi.itchihuahua.edu.mx/electro/archivo/electro2001/mem2001/articulos/cmp6.pdf
- Parra Castrillón, Eucario - Sistemas Tutoriales Inteligentes, Un Aporte de la Inteligencia Artificial para la Mediación Pedagógica - Versión Electrónica <En Línea> 2004. URL: <http://www.alfa.une.edu.ve/biblio/BiblioGeneral/S/S/sistemastutoriales.asp>
- Polson MC, Richardson JJ. Foundations of Intelligent Tutoring Systems (NJ): Lawrence Erlbaum Associates Inc. 1988.
- Revista Digital Matemática, Educación e Internet – Versión Electrónica <En Línea> 2005. URL:

<http://www.cidse.itcr.ac.cr/revistamate/HERRAInternet/ecuaexecl/node6.html>

- Sánchez Ilabaca, Jaime – Aprender Interactivamente con los Computadores – Versión Electrónica <En Línea> 1999. URL: <http://www.c5.cl/ie/ie/articulos/aprender.html>
- Santana Tejero, Ruth. DENDRAL: El primer sistema experto basado en conocimiento. Versión Electrónica <En Línea> 2004. URL: <http://www.it.uc3m.es/jvillena/irc/practicas/estudios/DENDRAL.pdf>
- Sierra, García-Martínez y Cataldi - Fundamentos para una metodología de diseño de sistemas tutoriales inteligentes centrada en la reparación de mecanismos – Versión Electrónica <En Línea> 2004. URL: <http://www.itba.edu.ar/capis/webcapis/RGMITBA/comunicacionesrgm/cacic2004-metodologia-diseno-sistemas-tutoriales-inteligentes.pdf>
- Tapia, Gustavo. Análisis Numérico. Raíces de Ecuaciones No Lineales – Versión Electrónica <En Línea> 2001. URL: <http://docentes.uacj.mx/gtapia/AN/Unidad2/Unidad2.htm>
- Turing, Alan M. – Computing Machinery and Intelligence. 1950. – Versión Electrónica <En Línea> 2001. URL: <http://www.abelard.org/turpap/turpap.php>
- Turing, Alan M. – On Computable Numbers, with an application to Entscheidungsproblem. 1936. – Versión Electrónica <En Línea> 2001. URL: <http://www.abelard.org/turpap2/tp2-ie.asp>

- Universidad de Deusto. Educación. Una incógnita en el cambio actual. Bilbao, 1998. Pág. 154.
- Urretavizcaya, Maite – Sistemas Inteligentes en el ámbito de la Educación. Versión Electrónica. <En Línea> 2001. URL: <http://aepia.dsic.upv.es/revista/numeros/12/Urretavizcaya.pdf>

A N E X O S

ANEXO A

Documento de diseño

Leonardo José Reyes Diago

Andrés Darío De la Ossa Arroyo

{ljreyesd, andros89}@gmail.com

Ingenieros de Sistemas

Universidad Tecnológica de Bolívar

Cartagena de Indias, D. T. y C.

2008

Sistema Tutor Inteligente de Asistencia a la Enseñanza de Métodos Numéricos
de Ingeniería Aplicados a la Resolución de Ecuaciones Algebraicas No
Lineales.

Contenidos

Introducción.....	1
1. Requerimientos del sistema	2
1.1. Requerimientos funcionales	2
1.1.1. Definición de actores	2
1.1.2. Requerimientos del sistema descritos en lenguaje natural.....	3
1.1.3. Requerimientos del sistema basados en casos de uso	7
1.2. Requerimientos no funcionales	37
2. Modelo dinámico del sistema	38
2.1. Diagramas de secuencia	38
2.2. Diagrama de clases.....	61
2.3. Diagramas de casos de uso	63
2.3.1. Casos de uso de alumno	63
2.3.2. Casos de uso de profesor	64
3. Diseño del sistema	65
3.1. Descomposición en subsistemas	65
3.2. Descripción de subsistemas	67
3.2.1. Gestión de usuarios.....	67
3.2.2. Modelo del dominio	67
3.2.3. Modelo del tutor.....	67
3.2.4. Modelo pedagógico	68
3.2.5. Modelo del alumno	68
3.3. Correspondencia entre hardware y software	69
3.3.1. Descripción de componentes	69
3.3.2. Despliegue físico de componentes.....	76

3.4. Administración de datos	78
3.5. Control de acceso.....	79
3.6. Flujo de control	79
3.7. Condiciones de frontera.....	79
4. Modelado de datos.....	81
4.1. Estructura de la base de datos	81
4.1.1. Estructuras de parametrización	81
4.1.2. Estructuras asociadas a información de usuario	81
4.1.3. Estructuras asociadas a organización de conocimiento	82
4.1.4. Estructuras que entrelazan información de usuario y organización de conocimiento	86
4.2. Modelo entidad-relación de la base de datos	88
5. Licenciamiento	89
5.1. GNU Lesser General Public License	89
5.2. Licencia Pública General Reducida GNU	96

Introducción

Este documento recopila las etapas de diseño de un sistema tutor inteligente de asistencia a la enseñanza de métodos numéricos de ingeniería aplicados a la resolución de ecuaciones algebraicas no lineales.

En él se encuentran consignados una descripción detallada de los requerimientos funcionales y no funcionales con los que debe contar, su representación en diagramas UML para modelar su arquitectura y el flujo de eventos entre componentes del sistema.

Se recopilan también los aspectos generales de diseño y el modelado de los datos que circularán entre los componentes del sistema.

Este documento es parte de un sistema de software licenciado bajo GNU LGPL (Licencia Pública General Reducida de GNU), por lo que es software libre de código abierto, susceptible a mejoras siempre que se reconozca a los autores de este proyecto y a la licencia de software empleada.

1. Requerimientos del sistema

1.1. Requerimientos funcionales

1.1.1. Definición de actores

Id.	Nombre	Descripción
AC-1	Alumno	Representa al usuario, objeto de enseñanza. Es quién interactúa la mayor parte del tiempo con el sistema.
AC-2	Profesor	Es el responsable de realizar el mantenimiento de los contenidos de enseñanza (temas, niveles, nodos, preguntas) y revisar constantemente la progresión del aprendizaje de cada alumno.

1.1.2. Requerimientos del sistema descritos en lenguaje natural

Se quiere implementar un sistema de tutoría inteligente que asista a la enseñanza de métodos numéricos de ingeniería para la resolución de ecuaciones algebraicas no lineales, para lo cual se desea que satisfaga los siguientes requerimientos:

El uso del sistema debe estar disponible a través de la Web, para facilitar la enseñanza fuera de un aula de clase y que esta sea posible para cualquier persona sin que importe su ubicación geográfica. Debe ser fácil de descargar a través de cualquier navegador Web y fácil de ejecutar para cualquier sistema operativo.

El usuario del sistema, profesor o alumno, debe iniciar sesión para obtener los servicios del sistema tutor. Para ello debe ingresar una dirección de correo electrónico y una contraseña, de estar registrado en el sistema. En caso contrario, podrá crearse un nuevo usuario con la información requerida: nombres, apellidos, una dirección de correo electrónico, una contraseña y nuevamente la contraseña para confirmación.

Cada usuario está en libertad de modificar su información personal, para lo cual el sistema debe posibilitar esto. El usuario podrá cerrar la sesión de trabajo en el sistema cuando lo desee.

Un usuario podrá también darse de baja en el sistema cuando lo desee, para lo cual el sistema deberá solicitar confirmación del mismo.

El sistema debe estar en capacidad de verificar si las contraseñas dadas son correctas en el momento crear usuarios y/o modificar contraseñas y, de verificar la validez de una dirección de usuario y contraseña para iniciar una sesión.

El sistema debe segregar funcionalidades según sea el perfil del usuario que inicia sesión. Un usuario alumno, sólo podrá acceder a una sesión de enseñanza. Un usuario profesor, debe identificarse como administrador del sistema y es el único que puede realizar mantenimiento a las estructuras de conocimiento necesarias para una sesión de enseñanza.

Un usuario alumno, podrá iniciar una sesión de enseñanza, para lo cual el sistema debe estar en capacidad de almacenar cuál fue el último tema objeto de enseñanza, en caso de ingresar reiteradamente al sistema para ofrecer una explicación del mismo o una nueva sesión de comprobación de conocimiento.

En caso de que un alumno ingrese por primera vez al sistema, deberá formularsele una prueba preliminar para seleccionar una primera prueba de comprobación de conocimiento. En todo caso, al ingresar por primera vez a un tema de conocimiento, el sistema deberá ofrecer una explicación o presentación de contenidos del tema en curso. No obstante, el alumno podrá estar en capacidad de aceptarla o no para iniciar una sesión de preguntas.

El alumno podrá navegar entre contenidos, sólo si ya ha superado las comprobaciones de conocimiento respectivas por cada tema. No obstante, si navega a un tema ya visto, no podrá formularse de nuevo una sesión de preguntas.

El sistema debe estar en capacidad de evaluar las respuestas del alumno, para lo cual deberá contar con un componente que se encargue de esta función y posibilite la selección de nuevas unidades de comprobación de conocimiento según sea el desempeño del alumno. El sistema debe almacenar un guarismo de la obtención de conocimiento del alumno por cada unidad de comprobación que visite.

El sistema debe ofrecer al usuario profesor la capacidad de hacer mantenimiento a las estructuras de conocimiento necesarias para impartir una sesión de enseñanza a un usuario alumno. Son ellas en su orden: temas, contenidos a visualizar, niveles de dificultad, nodos o unidades de comprobación, preguntas a formular y retroalimentaciones. El sistema deberá estar en la capacidad de verificar la información al momento de crear, modificar e inclusive eliminar cualquiera de estas estructuras y ante todo preservar la integridad de la información.

El sistema deberá ofrecer al usuario profesor, la funcionalidad de consultar el historial de un alumno, para lo cual, el profesor deberá seleccionarlo de un listado

con lo cual quedará definida la consulta. Una vez hecho esto, el sistema arrojará la información correspondiente: Nodo Visitado, Nodo Anterior Visitado, Nodo Siguiente a Visitar y Eficiencia Obtenida en Nodo. Los resultados deben ser visualizados en una tabla y estar disponibles para exportación a un archivo o para impresión.

1.1.3. Requerimientos del sistema basados en casos de uso

RF01 – Inicialización de programa cliente

RF01		Inicialización de programa cliente.
Actores		Alumno.
Precondición		Ninguna.
Flujo de eventos	1	Alumno ingresa a la página Web del proyecto a través de un navegador Web.
	2	Alumno accede al programa cliente, pulsando el vínculo 'Iniciar Cliente'.
	3	El sistema inicia la descarga del programa cliente.
	4	Tras finalizar descarga, el programa cliente se activa y establece comunicación con el servicio Web del sistema tutor.
	5	El sistema tutor establece comunicación con el sistema gestor de la base de datos.
Post-condición		El alumno ha accedido con éxito al programa cliente.
Excepciones	3.1.	Si existe un fallo en la comunicación entre el equipo cliente y el servidor del sistema tutor, el sistema tutor abortará la descarga del programa cliente.

RF02 – Inicio de Sesión

RF02	Inicio de Sesión	
Actores	Alumno, Profesor	
Precondición	Haber realizado el caso de uso RF01.	
Flujo de eventos	1	El usuario activa la opción 'Iniciar Sesión' en el menú del programa cliente.
	2	El cliente muestra la ventana de inicio de sesión.
	3	El usuario ingresa una dirección de correo electrónico y una contraseña en la ventana de inicio de sesión.
	4	El usuario pulsa el botón 'OK'.
	5	El cliente envía al sistema tutor la información ingresada en la ventana de inicio de sesión.
	6	El sistema tutor corrobora que la dirección de correo electrónico y la contraseña ingresadas sean válidas.
	7	El sistema tutor obtiene el número de identificación del usuario.
	8	El sistema emite un mensaje en el cliente notificando que la sesión fue iniciada correctamente.
	9	El sistema muestra la ventana de bienvenida en el cliente.
Post-condición	El usuario ha iniciado sesión en el sistema tutor.	
Excepciones	6.1.	Si hace falta información necesaria, el sistema emite un mensaje de error informando este suceso y retorna a la ventana de inicio de sesión.
	6.2.	Si la dirección de correo electrónico ingresada no está registrada, el sistema emite un mensaje de error informando este suceso y retorna a la ventana de inicio de sesión.
	6.3.	Si la contraseña no coincide con la registrada para la dirección de correo electrónico, el sistema emite un mensaje de error notificando este hecho y retorna a la ventana de inicio de sesión.

RF03 – Sesión de enseñanza

RF03	Sesión de enseñanza	
Actores	Alumno	
Precondición	Haber realizado el caso de uso RF02.	
Flujo de eventos	1	El alumno activa la opción 'Sesión de enseñanza', en el menú del programa cliente.
	2	El cliente activa la ventana de sesión de enseñanza.
	3	El cliente envía al sistema tutor un mensaje indicando que el alumno ha activado esta opción.
	4	El sistema tutor activa la máquina de inferencia.
	5	El sistema tutor consulta en el historial de alumnos, el último nodo alcanzado para la identificación del alumno.
	6	El sistema tutor recopila la información inherente al nodo seleccionado. (Tema, nivel)
	7	El sistema tutor prepara la ventana de sesión de enseñanza del cliente según valor de último nodo alcanzado.
Post-condición	El usuario ha activado una sesión de enseñanza.	
Excepciones	5.1.	Si el sistema tutor no encuentra datos para el identificador del alumno, envía un mensaje a la máquina de inferencia para que le indique el nodo a seguir.
	7.1.	Si el nodo seleccionado es el primero que visitará el alumno para el tema que lo contiene, el sistema tutor prepara la ventana de sesión de enseñanza para una explicación o presentación de contenidos.
	7.2.	Si el nodo seleccionado no es el primero que visitará el alumno para el tema que lo contiene, el sistema tutor prepara la ventana de sesión de enseñanza para una sesión de preguntas y respuestas.

RF04 – Presentación de contenidos

RF04		Presentación de contenidos
Actores		Alumno
Precondición		Haber cumplido con el caso de uso RF04.
Flujo de eventos	1	El sistema tutor, según el tema que corresponda el nodo seleccionado, carga la dirección URL del tema a presentar.
	2	El sistema tutor envía al cliente, la información del tema que se presentará en pantalla (identificador, URL)
	3	El cliente carga a modo de página Web la URL del contenido del tema en la ventana de sesión de enseñanza.
	4	El alumno navega entre contenidos pulsando los botones 'Anterior' y 'Siguiente'
	5	En la última página de contenidos, el alumno activa la opción 'Iniciar Preguntas'.
Post-condición		El alumno ha estudiado los contenidos de un tema.
Excepciones	4.1.	El alumno puede navegar entre contenidos, cuantas veces lo considere necesario.

RF05 – Sesión de preguntas

RF05	Sesión de preguntas	
Actores	Alumno	
Precondición	Haber cumplido los casos de uso RF03 o RF04.	
Flujo de eventos	1	El sistema tutor consulta la información inherente del nodo vigente (Preguntas, Identificador de Pregunta, Enunciado de pregunta, Opciones de selección, Respuestas Posibles) en la base de datos y las carga en memoria.
	2	El sistema tutor muestra en la ventana de sesión de enseñanza, por pregunta asociada al nodo, el enunciado y las opciones de selección de la respuesta.
	3	El alumno selecciona las opciones que considere como respuestas correctas por pregunta.
	4	El alumno pulsa el botón 'Enviar Respuestas'.
	5	El cliente, agrupa por preguntas, las opciones seleccionadas por el alumno y las envía al sistema tutor, junto al identificador del nodo vigente.
	6	El sistema tutor recibe los datos enviados por el cliente.
	7	El sistema tutor, por pregunta, envía la identificación del usuario, el identificador de la pregunta, y las opciones seleccionadas por el usuario a la máquina de inferencia.
	8	La máquina de inferencia recibe los datos, realiza inferencias y cálculos de eficiencia por pregunta.
	9	La máquina de inferencia envía al sistema tutor, la identificación de retroalimentación según la respuesta del alumno, y la eficiencia por pregunta.
	10	El sistema tutor recibe los datos enviados por la máquina de inferencia, acumula la eficiencia para el nodo y los almacena en memoria.
	11	El sistema tutor envía un mensaje a la máquina de

		inferencia, indicando que no hay más preguntas asociadas al nodo vigente.
	12	La máquina de inferencia recibe el mensaje y realiza inferencia.
	13	La máquina de inferencia envía el identificador del nodo siguiente y el estado para preparación de la ventana de sesión de enseñanza.
	14	El sistema tutor actualiza el historial del alumno, para el identificador del alumno, identificador del nodo vigente, el nodo siguiente y la eficiencia obtenida en el nodo vigente.
	15	El sistema tutor inserta un nuevo registro en el historial del alumno con los siguientes valores: <ul style="list-style-type: none"> ▪ Nodo Anterior = Nodo Vigente. ▪ Nodo Vigente = Nodo Siguiete según eficiencia del alumno. ▪ Nodo Siguiete = 0.
	16	El sistema tutor envía al cliente, las retroalimentaciones de cada pregunta y la eficiencia obtenida en el nodo vigente.
	17	El alumno pulsa el botón 'Continuar' en la ventana de sesión de enseñanza.
	18	El sistema tutor almacena para el identificador del alumno, el nodo siguiente señalado por la máquina de inferencia como nuevo nodo vigente.
	19	El sistema tutor prepara la ventana de sesión de enseñanza según el estado de preparación enviado por la máquina de inferencia.
Post-condición		El alumno ha completado una sesión de preguntas.
Excepciones	19.1.	Si el estado de preparación entregado por la máquina de inferencia es igual a uno (1), el sistema tutor termina la sesión de enseñanza por motivo Aprendizaje Completo y envía un mensaje al alumno notificando este suceso.
	19.2.	Si el estado de preparación entregado por la máquina

		de inferencia es igual a menos uno (-1), el sistema envía un mensaje de error informando al alumno que es necesaria una nueva explicación para el tema y prepara la ventana de sesión de enseñanza para ello.
	19.3.	Si el estado de preparación entregado por la máquina de inferencia es igual a cero (0), el sistema tutor prepara la ventana de sesión de enseñanza para una nueva sesión de preguntas.
	19.4.	Si el nodo siguiente entregado por la máquina de inferencia está asociado a un tema diferente al tema asociado al nodo actual, el sistema tutor envía un mensaje indicando que el tema se ha aprendido con éxito y prepara la ventana de sesión de enseñanza para una nueva presentación de contenidos.

RF06 – Creación de un alumno

RF06		Creación de un alumno
Actores		Alumno
Precondición		Haber realizado el caso de uso RF01.
Flujo de eventos	1	El alumno activa la opción 'Crear Usuario' en el menú del programa cliente.
	2	El programa cliente abre la ventana de creación de usuario en la que solicita del alumno sus datos personales (nombres, apellidos, nivel académico), una dirección de correo electrónico de ingreso, una contraseña y nuevamente la contraseña.
	3	El alumno ingresa la información solicitada y pulsa el botón 'Aceptar'.
	4	El cliente envía la información al sistema tutor.
	5	El sistema tutor recibe la información, la verifica y comprueba que las contraseñas coincidan.
	6	El sistema tutor registra al alumno como un usuario del sistema.
	7	El sistema tutor emite un mensaje notificando que el alumno ha sido registrado satisfactoriamente.
Post-condición		El alumno ha sido registrado como un usuario del sistema.
Excepciones	5.1.	Si hace falta información necesaria, el sistema emite un mensaje informando esto y retorna a la ventana de creación de usuario.
	5.2.	Si las contraseñas no coinciden, el sistema emite un mensaje informando esto y retorna a la ventana de creación de usuario.
	5.3.	Si la dirección de correo electrónico entregada, ya ha sido registrada en el sistema, se emitirá una notificación sobre esto y retornará a la página de creación de usuario.

RF07 – Modificar información personal de un alumno

RF07	Modificar información personal de un alumno	
Actores	Alumno	
Precondición	Haber cumplido el caso de uso RF02.	
Flujo de eventos	1	El alumno activa la opción 'Modificar información' en el menú del programa cliente.
	2	El cliente envía al sistema tutor la solicitud de obtener la información personal del alumno.
	3	El sistema tutor recibe la solicitud y consulta en la base de datos la información del alumno
	4	El sistema tutor envía la información del alumno al programa cliente.
	5	El cliente despliega la ventana de información personal del alumno con la información del mismo.
	6	El alumno ingresa sus nuevos datos personales su contraseña actual, su nueva contraseña y reiteradamente su contraseña para confirmación en la ventana de información del alumno.
	7	El alumno pulsa el botón 'OK'.
	8	El programa cliente envía la información ingresada en la ventana de información del alumno al sistema tutor.
	9	El sistema tutor corrobora que la información suministrada sea válida.
	10	El sistema actualiza los datos modificados.
	11	El sistema emite un mensaje informando que la información de alumno ha sido modificada con éxito.
Post-condición	La información del usuario se ha modificado.	
Excepciones	8.1.	Si hace falta información necesaria, el sistema emite un mensaje informando esto y retorna a la ventana de información de alumno.
	8.2.	Si la contraseña no coincide con la identificación del alumno, el sistema emite un mensaje informando

		esto y retorna a la ventana de información de alumno.
	8.3.	Si las contraseñas no coinciden, el sistema emite un mensaje informando esto y retorna a la ventana de información de alumno.

RF08 – Dar de baja a un alumno

RF08	Dar de baja a un alumno	
Actores	Alumno	
Precondición	Haber realizado el caso de uso RF02.	
Flujo de eventos	1	El alumno activa la opción 'Modificar información' en el menú del programa cliente.
	2	El alumno activa la opción 'Modificar información' en el menú del programa cliente.
	3	El cliente envía al sistema tutor la solicitud de obtener la información personal del alumno.
	4	El sistema tutor recibe la solicitud y consulta en la base de datos la información del alumno
	5	El sistema tutor envía la información del alumno al programa cliente.
	6	El cliente despliega la ventana de información personal del alumno con la información del mismo.
	7	El alumno pulsa el botón 'Dar de baja' en la ventana de información del alumno.
	8	El cliente solicita confirmación para la dada de baja del alumno.
	9	El alumno pulsa el botón 'OK'.
	10	El cliente envía la solicitud de dada de baja del alumno al sistema tutor.
	11	El sistema tutor recibe la solicitud y actualiza el estado del alumno a inactivo.
	12	El sistema tutor emite un mensaje informando que el alumno ha sido dado de baja.
	13	El sistema tutor finaliza la sesión del alumno.
Post-condición	El alumno ha sido dado de baja.	
Excepciones		

RF09 – Creación de temas

RF09		Creación de temas
Actores		Profesor
Precondición		Haber cumplido con el caso de uso RF02.
Flujo de eventos	1	El profesor activa la opción 'Mantenimiento de Temas' en el menú del programa cliente.
	2	El sistema activa la ventana 'Mantenimiento de Temas'.
	3	El profesor pulsa el botón 'Nuevo' para crear un nuevo tema.
	4	El sistema prepara la ventana y solicita del profesor una identificación para el tema, un nombre, los temas precedente y subsiguiente y su listado de direcciones URL de contenido de enseñanza.
	5	El profesor ingresa los datos solicitados y pulsa el botón 'Actualizar'.
	6	El sistema verifica los datos ingresados.
	7	El sistema crea un nuevo tema en la base de datos de contenidos.
	8	El sistema emite un mensaje notificando que el tema se ha creado exitosamente.
Post-condición		Se ha creado un nuevo tema.
Excepciones	6.1.	Si hace falta información requerida, el sistema emite un mensaje y retorna a la ventana de mantenimiento de temas.
	6.2.	Si la identificación del tema ingresada ya existe en la base de datos, el sistema emite un mensaje de error y retorna a la ventana de mantenimiento de temas.
	6.3.	Si los temas precedente y subsiguiente no existen, el sistema emite un mensaje y retorna a la ventana de mantenimiento de temas.
	6.4.	Si no se ha registrado al menos una dirección URL de contenidos para el tema, el sistema emite un mensaje de error y retorna a la ventana de

		mantenimiento de temas.
--	--	-------------------------

RF10 – Modificación de temas

RF10	Modificación de temas	
Actores	Profesor	
Precondición	Haber cumplido con el caso de uso RF02.	
Flujo de eventos	1	El profesor activa la opción 'Mantenimiento de Temas' en el menú del programa cliente.
	2	El sistema activa la ventana 'Mantenimiento de Temas'
	3	El profesor selecciona de una lista desplegable, el tema del que desea obtener información.
	4	El sistema carga en la ventana la información del tema seleccionado.
	5	El profesor modifica la información que considere pertinente y pulsa el botón 'Actualizar'.
	6	El sistema verifica los datos ingresados.
	7	El sistema actualiza la información para el tema en la base de datos.
	8	El sistema emite un mensaje notificando que el tema se ha modificado exitosamente.
Post-condición	El sistema actualiza la información del tema.	
Excepciones	6.1.	Si hace falta información requerida, el sistema emite un mensaje y retorna a la ventana de mantenimiento de temas.
	6.2.	Si los temas precedente y subsiguiente no existen, el sistema emite un mensaje y retorna a la ventana de mantenimiento de temas.
	6.3.	Si no se ha registrado al menos una dirección URL de contenidos para el tema, el sistema emite un mensaje de error y retorna a la ventana de mantenimiento de temas.

RF11 – Eliminación de temas

RF11	Eliminación de temas	
Actores	Profesor	
Precondición	Haber cumplido con el caso de uso RF02.	
Flujo de eventos	1	El profesor activa la opción 'Mantenimiento de Temas' en el menú del programa cliente.
	2	El sistema activa la ventana 'Mantenimiento de Temas'
	3	El profesor selecciona de una lista desplegable, el tema del que desea obtener información.
	4	El sistema carga en la ventana la información del tema seleccionado.
	5	El profesor pulsa el botón 'Eliminar'.
	6	El sistema notifica que la acción es irreversible puesto que toda la información asociada al tema también se eliminará y solicita confirmación para proceder.
	7	El profesor pulsa el botón 'OK'.
	8	El sistema verifica la integridad de los datos.
	9	El sistema elimina el tema de los contenidos de enseñanza, sus niveles de dificultad y sus preguntas asociadas.
	10	El sistema emite un mensaje notificando que el tema ha sido eliminado con éxito.
Post-condición	El sistema ha eliminado el tema de los contenidos de enseñanza.	
Excepciones	9.1.	Si el tema es subsiguiente o precedente de otro, el sistema emite un mensaje de error, notificando esto y sugiriendo que desligar el tema.

RF12 – Mantenimiento de niveles

RF12		Mantenimiento de niveles
Actores		Profesor
Precondición		Haber cumplido con el caso de uso RF02.
Flujo de eventos	1	El profesor activa la opción 'Mantenimiento de Temas' en el menú del programa cliente.
	2	El sistema activa la ventana 'Mantenimiento de Temas'
	3	El profesor selecciona de una lista desplegable, el tema del que desea obtener información.
	4	El sistema carga en la ventana la información del tema seleccionado.
	5	El profesor pulsa el botón 'Niveles'.
	6	El sistema activa la ventana 'Mantenimiento de Niveles' y carga todos los niveles asociados al tema en la tabla 'Niveles'.
Post-condición		Se ha activado el mantenimiento de niveles.
Excepciones		

RF13 – Creación de niveles de dificultad

RF13	Creación de niveles de dificultad	
Actores	Profesor	
Precondición	Haber cumplido el caso de uso RF12.	
Flujo de eventos	1	El profesor pulsa el botón 'Nuevo Nivel' en la ventana 'Mantenimiento de Niveles'.
	2	El sistema inserta una nueva línea en la tabla 'Niveles' y solicita del profesor el rango de permisividad para la eficiencia de preguntas y la tolerancia.
	3	El profesor ingresa los datos solicitados y pulsa el botón 'Actualizar'.
	4	El sistema verifica la validez de los datos ingresados.
	5	El sistema crea un nuevo nivel para el tema.
	6	El sistema emite un mensaje informando que la creación del nivel se ha realizado con éxito.
Post-condición	Se ha creado un nuevo nivel.	
Excepciones	4.1.	Si hace falta información necesaria, el sistema emite un mensaje de error y retorna a la ventana 'Mantenimiento de Niveles'.
	4.2.	Si el límite inferior del rango de permisividad es mayor que el límite superior, el sistema emite un mensaje de error y retorna a la ventana 'Mantenimiento de Niveles'.
	4.3.	Si el valor ingresado para la tolerancia es menor que cero, el sistema emite un mensaje de error y retorna a la ventana 'Mantenimiento de Niveles'.

RF14 – Modificación de niveles de dificultad

RF14	Modificación de niveles de dificultad	
Actores	Profesor	
Precondición	Haber cumplido el caso de uso RF12.	
Flujo de eventos	1	El profesor selecciona el nivel al que desea modificar información en la ventana 'Mantenimiento de Niveles'.
	2	El profesor modifica la información que considere pertinente sobre la línea seleccionada.
	3	El profesor ingresa los datos solicitados y pulsa el botón 'Actualizar Nivel'.
	4	El sistema verifica la validez de los datos ingresados.
	5	El sistema la información para el nivel seleccionado.
	6	El sistema emite un mensaje informando que la modificación del nivel se ha realizado con éxito.
Post-condición	Se ha modificado la información de un nivel.	
Excepciones	4.1.	Si hace falta información necesaria, el sistema emite un mensaje de error y retorna a la ventana 'Mantenimiento de Niveles'.
	4.2.	Si el límite inferior del rango de permisividad es mayor que el límite superior, el sistema emite un mensaje de error y retorna a la ventana 'Mantenimiento de Niveles'.
	4.3.	Si el valor ingresado para la tolerancia es menor que cero, el sistema emite un mensaje de error y retorna a la ventana 'Mantenimiento de Niveles'.

RF15 – Eliminación de niveles de dificultad

RF15		Eliminación de niveles de dificultad
Actores		Profesor
Precondición		Haber cumplido el caso de uso RF12.
Flujo de eventos	1	El profesor selecciona el nivel que desea eliminar y pulsa el botón 'Eliminar Nivel' en la ventana 'Mantenimiento de Niveles'.
	2	El sistema notifica que la acción es irreversible puesto que toda la información asociada al nivel también se eliminará y solicita confirmación para proceder.
	3	El profesor pulsa el botón 'OK'.
	4	El sistema verifica la integridad de los datos.
	5	El sistema elimina el nivel de dificultad y sus preguntas asociadas.
	6	El sistema emite un mensaje notificando que el nivel ha sido eliminado con éxito.
Post-condición		Se ha eliminado un nivel de dificultad.
Excepciones		

RF16 – Creación de nodos

RF16		Creación de nodos
Actores		Profesor
Precondición		Haber cumplido el caso de uso RF12.
Flujo de eventos	1	El profesor selecciona el nivel al que desea modificar información en la ventana 'Mantenimiento de Niveles'.
	2	El sistema carga en la tabla 'Nodos', los nodos asociados al nivel seleccionado.
	3	El profesor pulsa el botón 'Nuevo Nodos'.
	4	El sistema inserta una nueva línea en la tabla 'Nodos' y solicita del profesor la dificultad del nodo de una lista desplegable.
	5	El profesor ingresa los datos solicitados y pulsa el botón 'Actualizar'.
	6	El sistema verifica la validez de los datos ingresados.
	7	El sistema crea un nuevo nodo para el nivel seleccionado.
	8	El sistema emite un mensaje informando que la creación del nodo se ha realizado con éxito.
Post-condición		Se ha creado un nuevo nodo.
Excepciones	6.1.	Si no se ha definido la dificultad para el nodo, el sistema emite un mensaje de error y retorna a la ventana 'Mantenimiento de Niveles'.

RF17 – Modificación de nodos

RF17		Modificación de nodos
Actores		Profesor
Precondición		Haber cumplido el caso de uso RF02.
Flujo de eventos	1	El profesor selecciona el nivel al que desea modificar información en la ventana 'Mantenimiento de Niveles'.
	2	El sistema carga en la tabla 'Nodos', los nodos asociados al nivel seleccionado.
	3	El profesor selecciona el nodo al que desea modificar información.
	4	El profesor modifica la información que considere pertinente y pulsa el botón 'Actualizar'.
	5	El sistema verifica los datos ingresados.
	6	El sistema actualiza la información del nodo seleccionado en la base de datos.
	7	El sistema emite un mensaje informando que la actualización ha sido exitosa
Post-condición		Se ha eliminado un nodo.
Excepciones	5.1.	Si no se ha definido la dificultad para el nodo, el sistema emite un mensaje de error y retorna a la ventana 'Mantenimiento de Niveles'.

RF18 – Eliminación de nodos

RF18		Eliminación de nodos
Actores		Profesor
Precondición		Haber cumplido con el caso de uso RF12.
Flujo de eventos	1	El profesor selecciona el nivel al que desea modificar información en la ventana 'Mantenimiento de Niveles'.
	2	El sistema carga en la tabla 'Nodos', los nodos asociados al nivel seleccionado.
	3	El profesor selecciona el nodo que desea eliminar.
	4	El profesor pulsa el botón 'Eliminar Nodo'.
	5	El sistema emite un mensaje advirtiendo que la operación a realizar es irreversible y que las preguntas asociadas al nodo seleccionado se perderán y pide confirmación para proceder.
	6	El profesor pulsa el botón 'OK'.
	7	El sistema elimina el nodo seleccionado y sus preguntas asociadas.
	8	El sistema emite un mensaje notificando que la eliminación se ha realizado con éxito.
Post-condición		Se ha eliminado un nodo.
Excepciones		

RF19 – Creación de preguntas

RF19	Creación de preguntas	
Actores	Profesor	
Precondición	Haber cumplido con el caso de uso RF12.	
Flujo de eventos	1	El profesor selecciona el nivel al que desea modificar información en la ventana 'Mantenimiento de Niveles'.
	2	El sistema carga en la tabla 'Nodos', los nodos asociados al nivel seleccionado.
	3	El profesor selecciona el nodo al que desea asociar preguntas.
	4	El profesor pulsa el botón 'Nueva Pregunta'.
	5	El sistema habilita el formulario donde solicita del profesor, el enunciado de la pregunta, el tipo de pregunta, la ponderación, el rango de eficiencia, las opciones de selección, las respuestas correctas.
	6	El profesor ingresa la información requerida.
	7	El profesor pulsa el botón 'Actualizar'.
	8	El sistema verifica la información ingresada por el profesor.
	9	El sistema crea la pregunta y sus opciones y prepara sus retroalimentaciones asociadas.
	10	El sistema muestra un mensaje informando que la pregunta se ha creado correctamente.
Post-condición	Se ha creado la pregunta con éxito.	
Excepciones	6.1.	Según sea el tipo de pregunta, cambia la petición de opciones.
	8.1.	Si hace falta información necesaria, el sistema envía un mensaje de error y retorna a la ventana 'Mantenimiento de Niveles'.
	8.2.	Si la ponderación de la pregunta es menor o igual a cero, el sistema emite un mensaje de error y retorna a la ventana 'Mantenimiento de Niveles'.
	8.3.	Si la suma de las ponderaciones de todas las

		preguntas asociadas al nodo son mayores que 100, el sistema emite un mensaje de error y retorna a la ventana 'Mantenimiento de Niveles'.
	8.4.	Si el límite inferior del rango de eficiencia es mayor que el límite superior, el sistema emite un mensaje de error y retorna a la ventana de 'Mantenimiento de Niveles'.

RF20 – Modificación de preguntas

RF20	Modificación de preguntas	
Actores	Profesor	
Precondición	Haber cumplido con el caso de uso RF12.	
Flujo de eventos	1	El profesor selecciona el nivel al que desea modificar información en la ventana 'Mantenimiento de Niveles'.
	2	El sistema carga en la tabla 'Nodos', los nodos asociados al nivel seleccionado.
	3	El profesor selecciona el nodo al que desea asociar preguntas.
	4	El sistema carga en la tabla 'Preguntas', las preguntas asociadas al nodo seleccionado.
	5	El profesor selecciona la pregunta a la que desea actualizar información.
	6	El sistema carga en el formulario dispuesto para editar información de preguntas, los datos de la pregunta seleccionada (el enunciado de la pregunta, el tipo de pregunta, la ponderación, el rango de eficiencia, las opciones de selección, las respuestas correctas).
	7	El profesor modifica la información que considere pertinente y pulsa el botón 'Actualizar'.
	8	El sistema verifica los datos suministrados.
	9	El sistema actualiza los datos asociados a la pregunta.
	10	El sistema emite un mensaje informando que la actualización se completado con éxito.
Post-condición		
Excepciones	6.1.	Según sea el tipo de pregunta, cambia la petición de opciones.
	8.1.	Si hace falta información necesaria, el sistema envía un mensaje de error notificando esto y retorna a la ventana 'Mantenimiento de Niveles'.

	8.2.	Si la ponderación de la pregunta es menor o igual a cero, el sistema emite un mensaje de error y retorna a la ventana 'Mantenimiento de Niveles'.
	8.3.	Si la suma de las ponderaciones de todas las preguntas asociadas al nodo son mayores que 100, el sistema emite un mensaje de error y retorna a la ventana 'Mantenimiento de Niveles'.
	8.4.	Si el límite inferior del rango de eficiencia es mayor que el límite superior, el sistema emite un mensaje de error y retorna a la ventana 'Mantenimiento de Niveles'.

RF21 – Eliminación de preguntas

RF21	Eliminación de preguntas	
Actores	Profesor	
Precondición	Haber cumplido el caso de uso RF12.	
Flujo de eventos	1	El profesor selecciona el nivel al que desea modificar información en la ventana 'Mantenimiento de Niveles'.
	2	El sistema carga en la tabla 'Nodos', los nodos asociados al nivel seleccionado.
	3	El profesor selecciona el nodo al que desea asociar preguntas.
	4	El sistema carga en la tabla 'Preguntas', las preguntas asociadas al nodo seleccionado.
	5	El profesor selecciona la pregunta que desea eliminar.
	6	El profesor pulsa el botón 'Eliminar Pregunta'.
	7	El sistema solicita confirmación para eliminar la pregunta.
	8	El profesor pulsa el botón 'OK'.
	9	El sistema elimina la pregunta y sus retroalimentaciones de la base de datos.
	10	El sistema emite una notificación informando que la pregunta fue eliminada con éxito.
Post-condición	Se ha eliminado la pregunta.	
Excepciones		

RF22 – Mantenimiento de retroalimentaciones

RF22		Mantenimiento de retroalimentaciones.
Actores		Profesor
Precondición		Haber cumplido con el caso de uso RF12.
Flujo de eventos	1	El profesor selecciona el nivel al que desea modificar información en la ventana 'Mantenimiento de Niveles'.
	2	El sistema carga en la tabla 'Nodos', los nodos asociados al nivel seleccionado.
	3	El profesor selecciona el nodo al que desea asociar preguntas.
	4	El sistema carga en la tabla 'Preguntas', las preguntas asociadas al nodo seleccionado.
	5	El profesor selecciona la pregunta a la que desea actualizar información.
	6	El sistema carga en el formulario dispuesto para editar información de preguntas, los datos de la pregunta seleccionada (el enunciado de la pregunta, el tipo de pregunta, la ponderación, el rango de eficiencia, las opciones de selección, las respuestas correctas).
	7	El profesor pulsa el botón 'Feedback'.
	8	El sistema activa la ventana 'Feedback' y carga las tres retroalimentaciones asociadas a la pregunta.
	9	El profesor edita las retroalimentaciones que considere pertinentes y pulsa el botón 'Actualizar'
	10	El sistema verifica los datos ingresados.
	11	El sistema actualiza las retroalimentaciones asociadas a la pregunta.
	12	El sistema emite un mensaje notificando que las retroalimentaciones se actualizaron con éxito.
Post-condición		Se han actualizado las retroalimentaciones.
Excepciones		

RF23 – Ver historial de un alumno

RF23		Ver historial de un alumno
Actores		Profesor
Precondición		Haber cumplido con el caso de uso RF02.
Flujo de eventos	1	El profesor activa la opción 'Ver Historial' en el menú del programa cliente.
	2	El sistema activa la ventana 'Historial de Alumno'.
	3	El sistema carga en una lista desplegable de la ventana 'Historial de Alumno', los alumnos registrados para ver su historial.
	4	El profesor selecciona de la lista el alumno del cual desea ver su historial.
	5	El sistema carga en una tabla en la ventana 'Historial de Alumno', la traza de seguimiento del alumno (Tema, Nodo, Nodo Anterior, Nodo Siguiente, Eficiencia Obtenida)
Post-condición		Se ha cargado el historial de un alumno.
Excepciones		

1.2. Requerimientos no funcionales

RNF01	Seguridad
Descripción	El sistema deberá ser seguro; cada usuario del sistema deberá tener su dirección de correo electrónico y su contraseña registrados. La contraseña debe almacenarse encriptada en MD5.

RNF02	Confiabilidad
Descripción	El sistema deberá ser confiable. No debe causar daños físicos o económicos en el caso de una falla en el sistema.

RNF03	Mantenibilidad
Descripción	El sistema deberá ser mantenible. Debe evolucionar para cumplir las necesidades de cambio.

RNF04	Adaptabilidad
Descripción	El sistema deberá ser adaptable a cualquier tipo de contenido de enseñanza.

RNF05	Portabilidad
Descripción	El sistema debe ser fácilmente portable a cualquier sistema operativo que soporte acceso a la Web.

RNF06	Mínimo 8 caracteres de contraseña
Descripción	La contraseña de un usuario del sistema deberá tener mínimo ocho (8) caracteres.

RNF07	Accesibilidad
-------	---------------

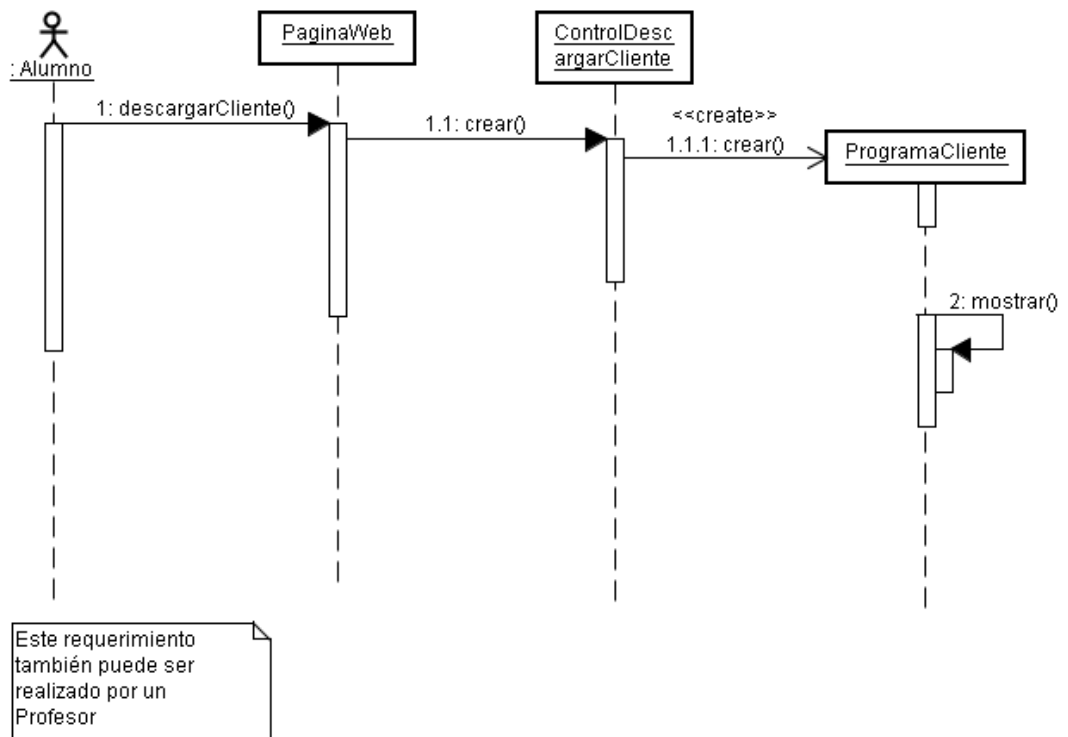
Descripción	El sistema deberá ser accesible todo el tiempo, 24 horas, 365 días.
-------------	---

RNF08	Concurrencia
Descripción	El sistema deberá soportar mínimo 20 usuarios concurrentes a través de la Web.

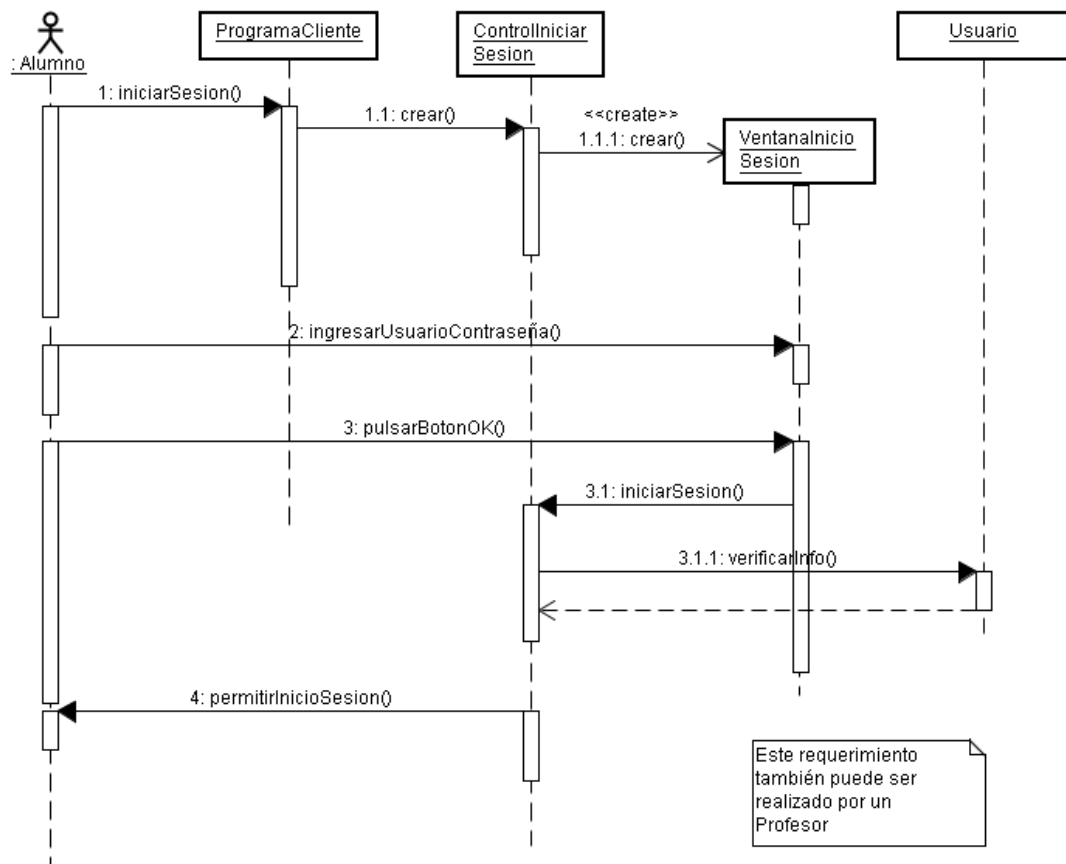
2. Modelo dinámico del sistema

2.1. Diagramas de secuencia

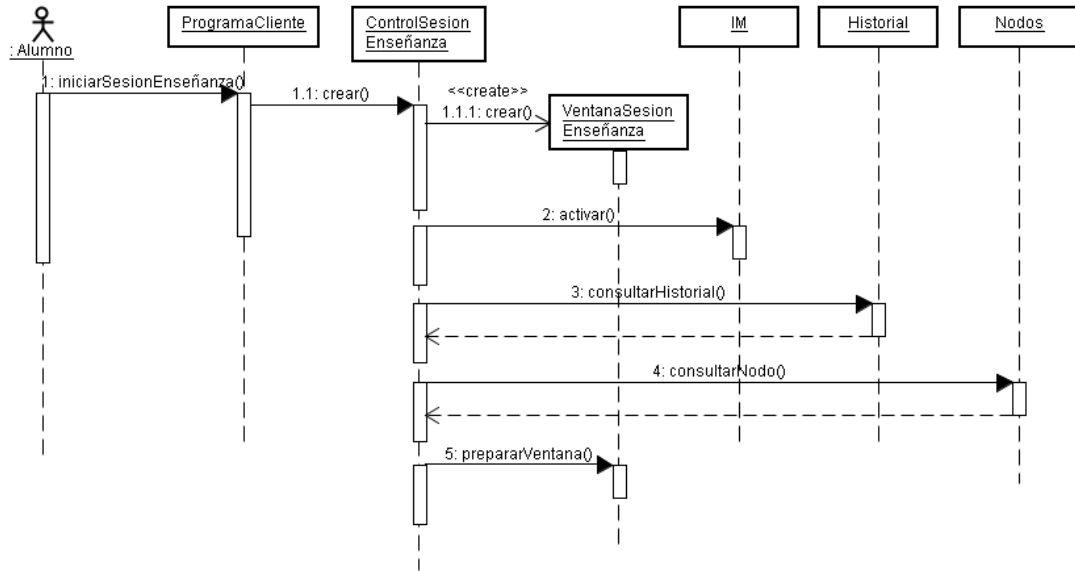
RF01 – Inicialización de programa cliente



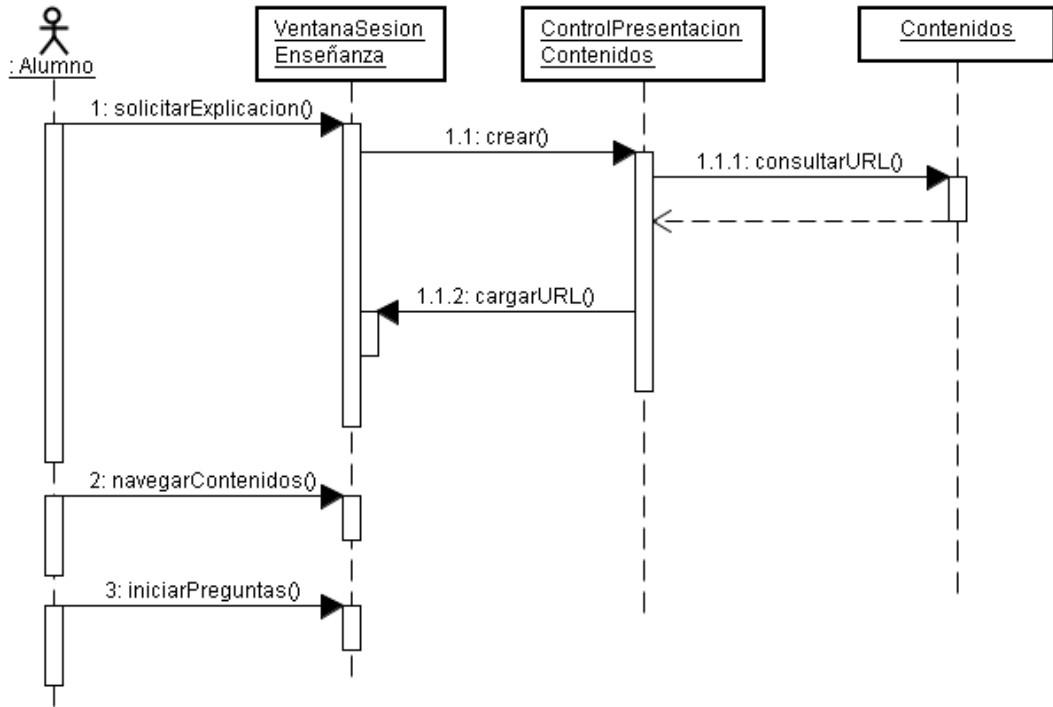
RF02 – Inicio de Sesión



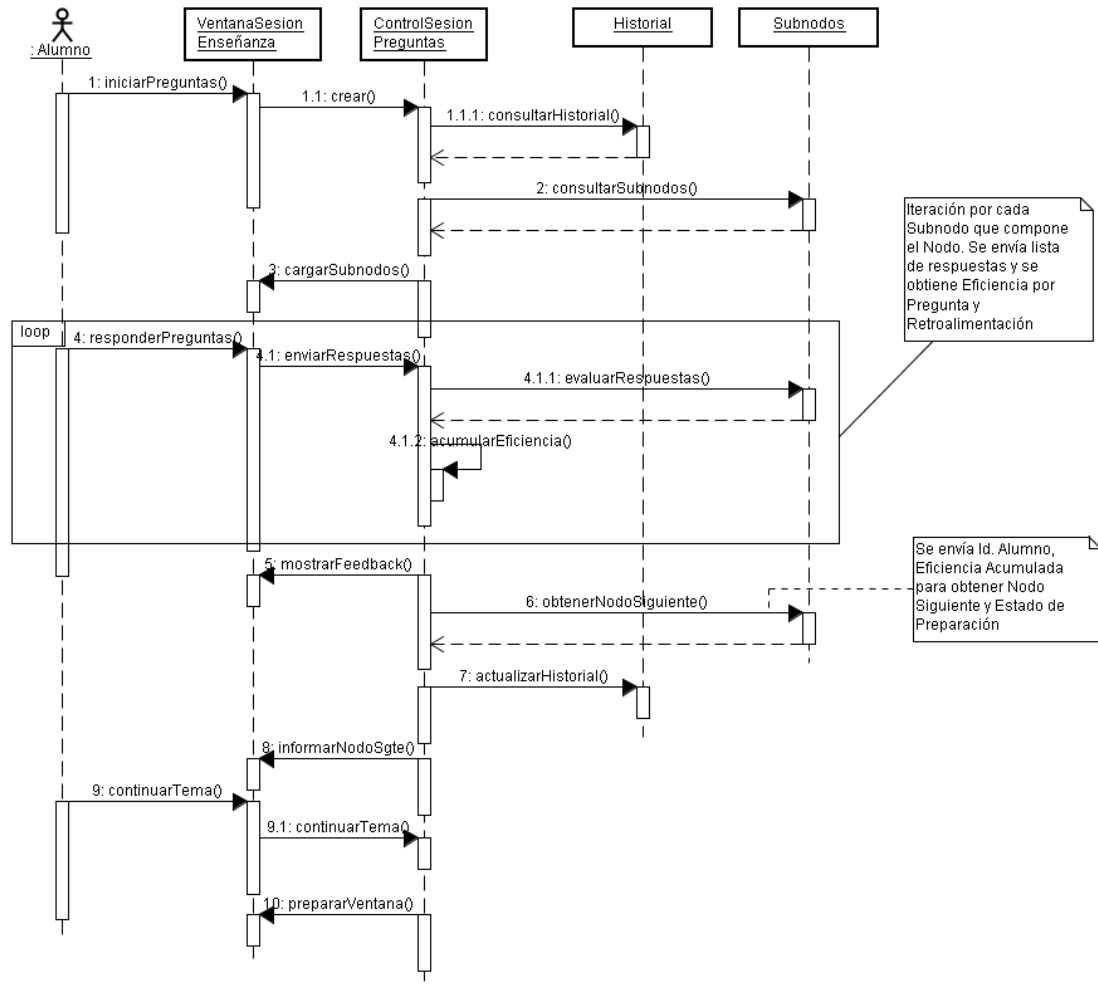
RF03 – Sesión de enseñanza



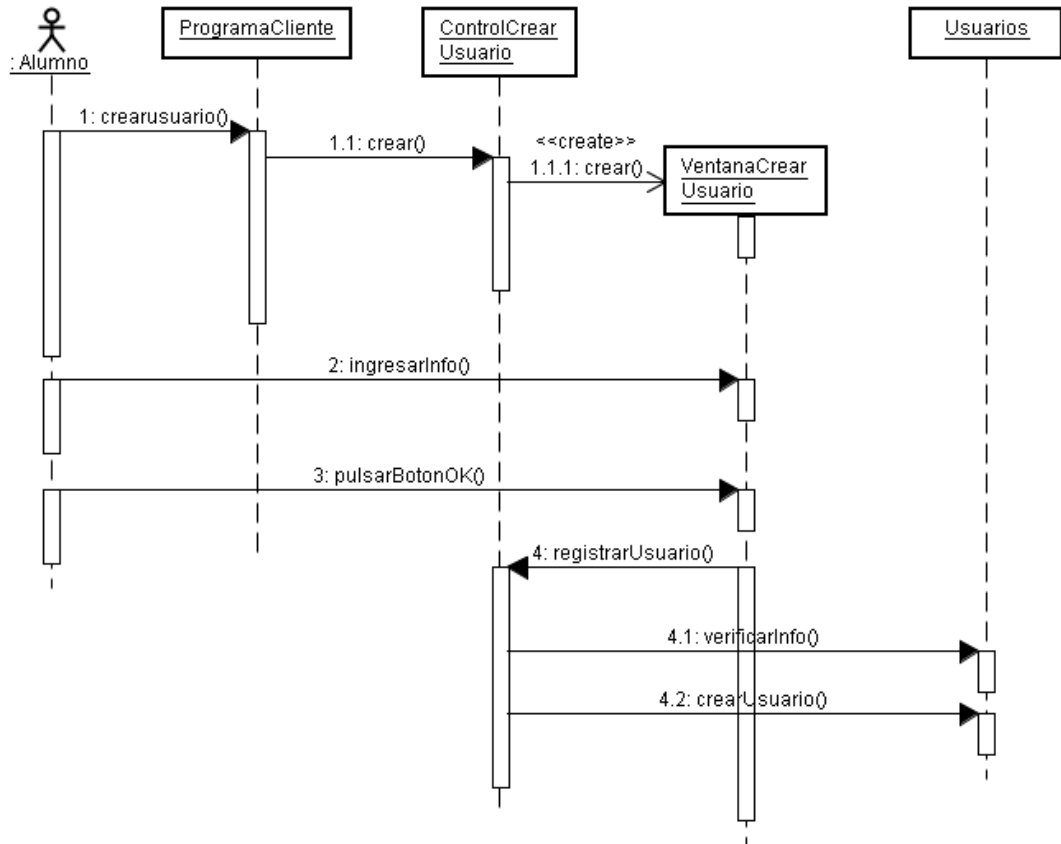
RF04 – Presentación de contenidos



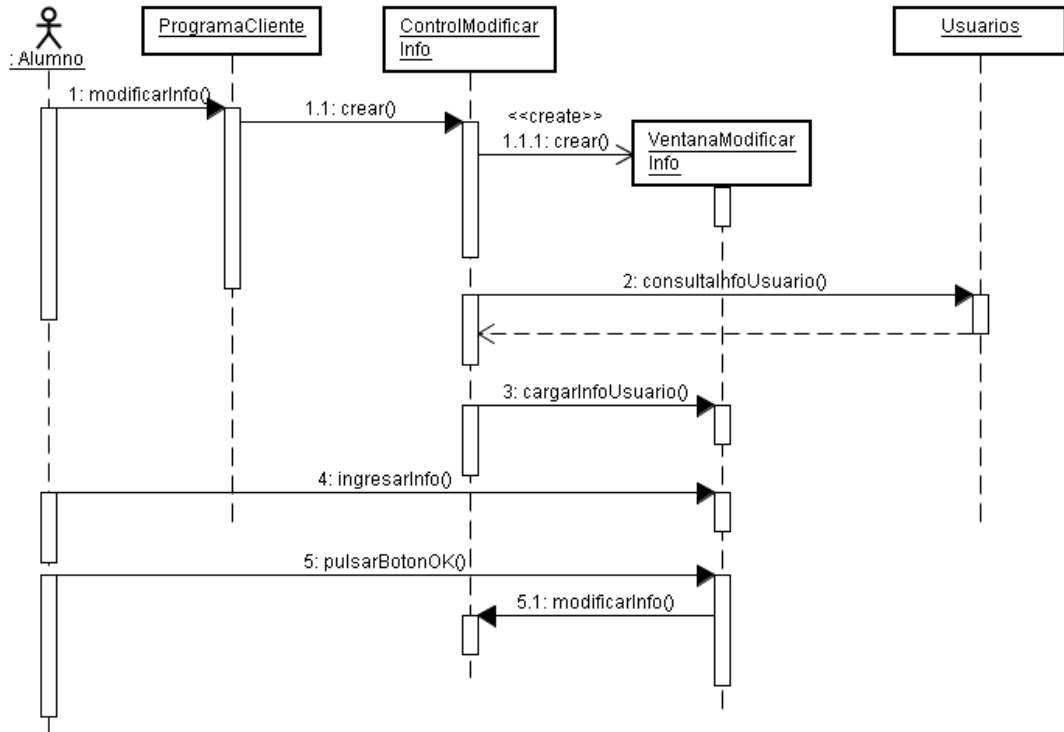
RF05 – Sesión de preguntas



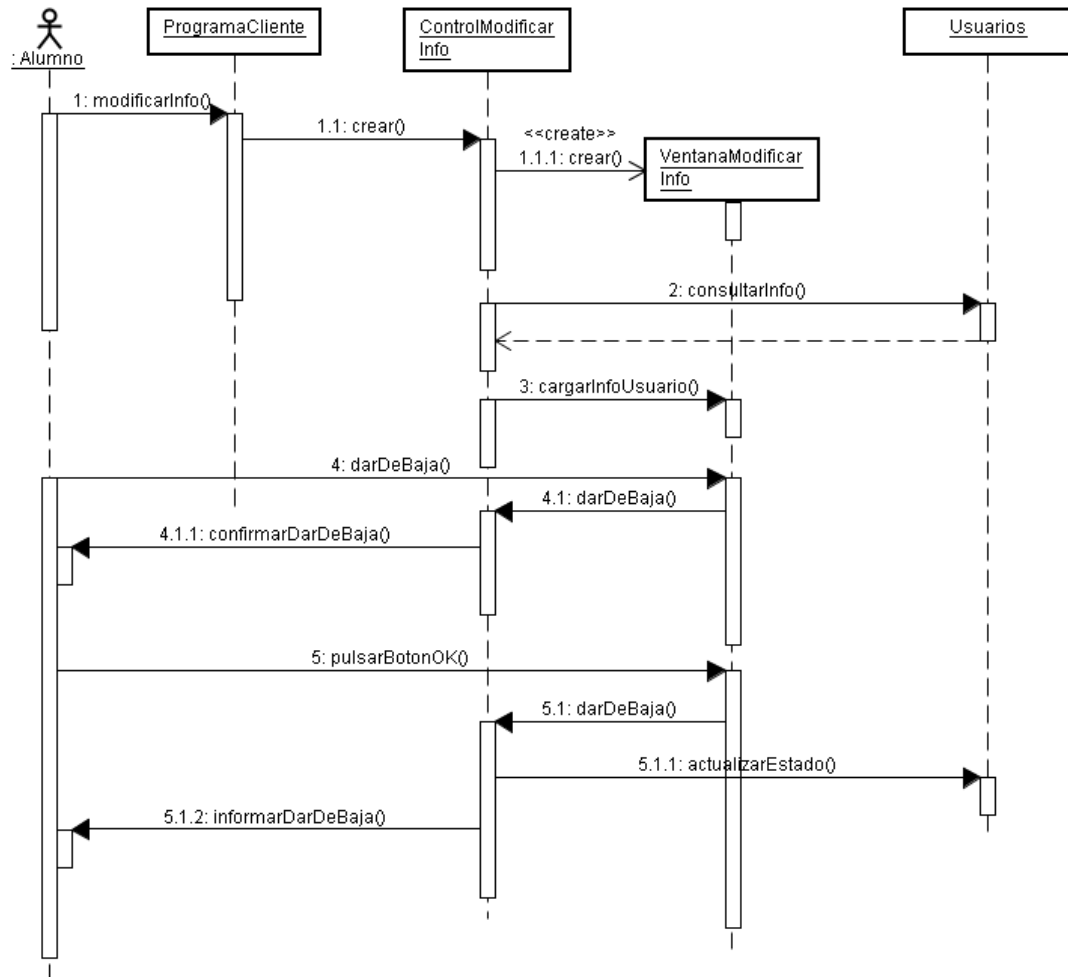
RF06 – Creación de un alumno



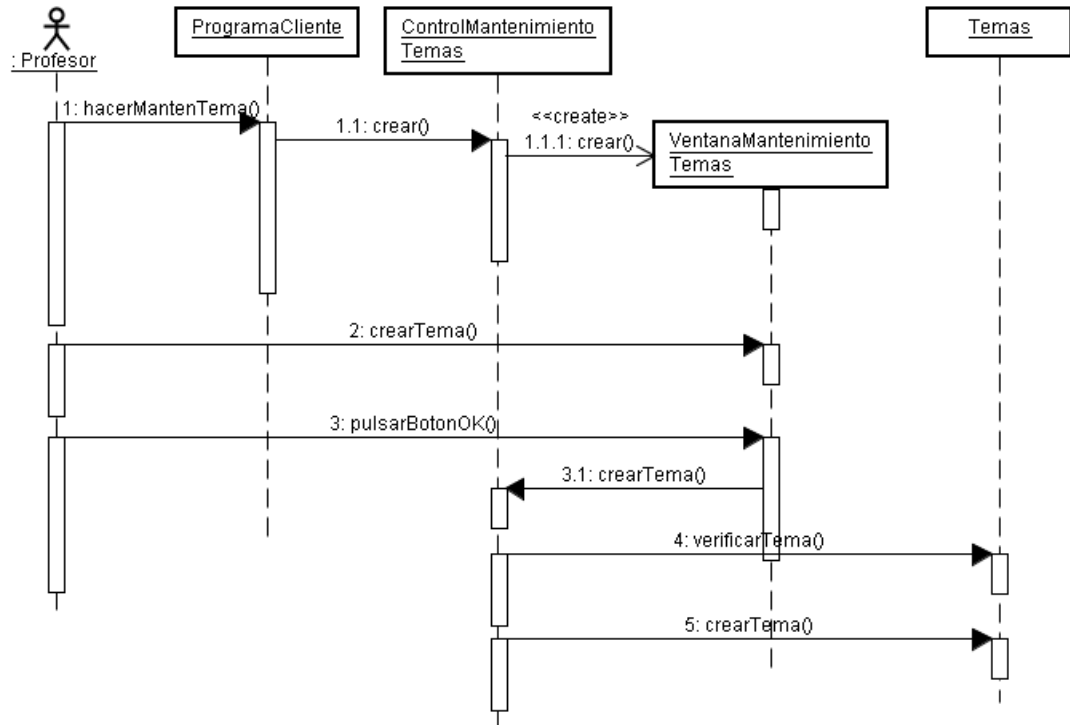
RF07 – Modificar información personal de un alumno



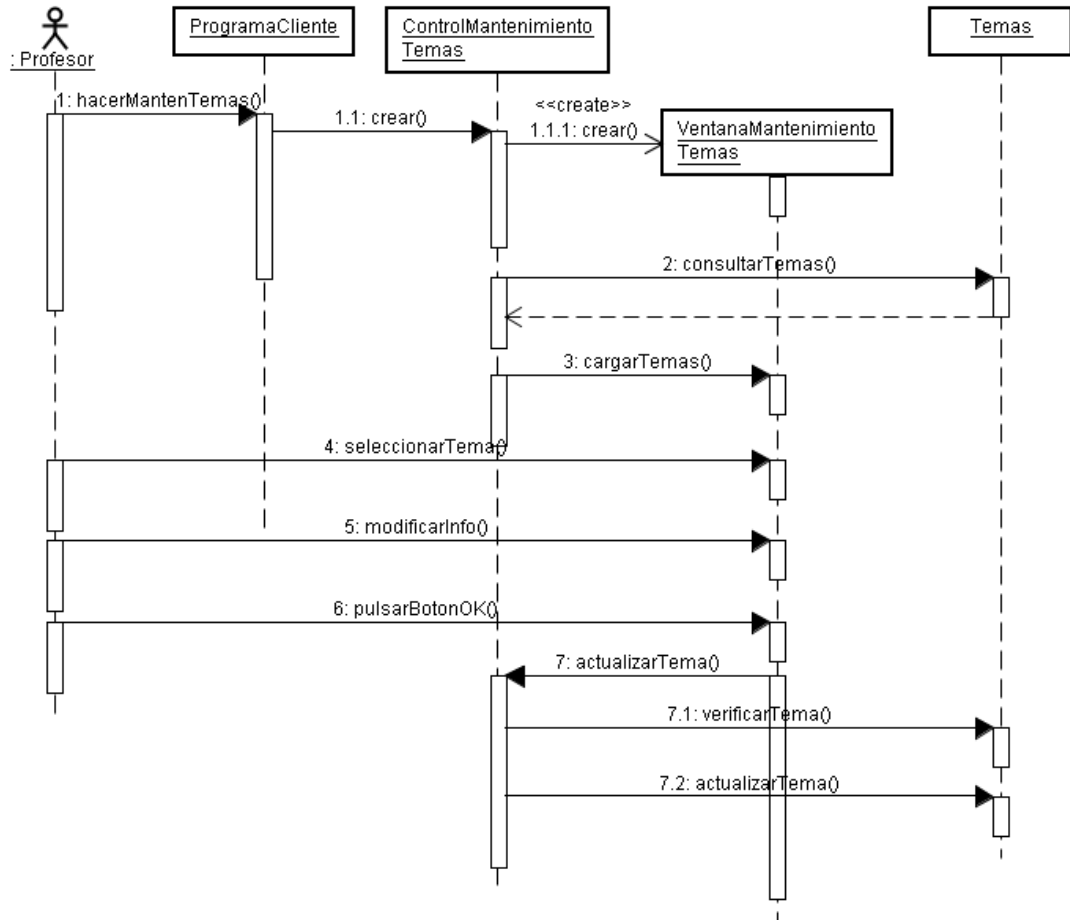
RF08 – Dar de baja a un alumno



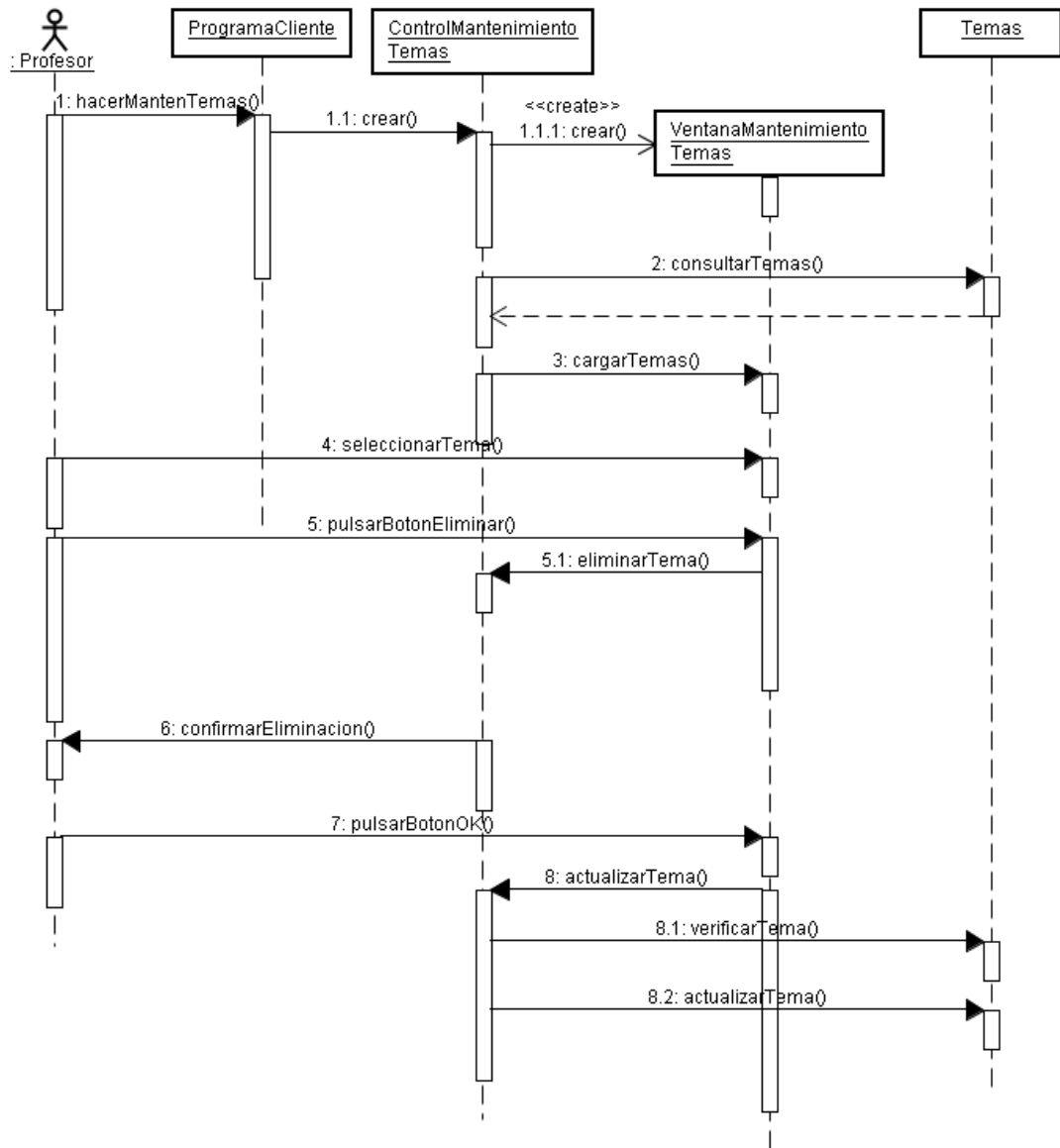
RF09 – Creación de temas



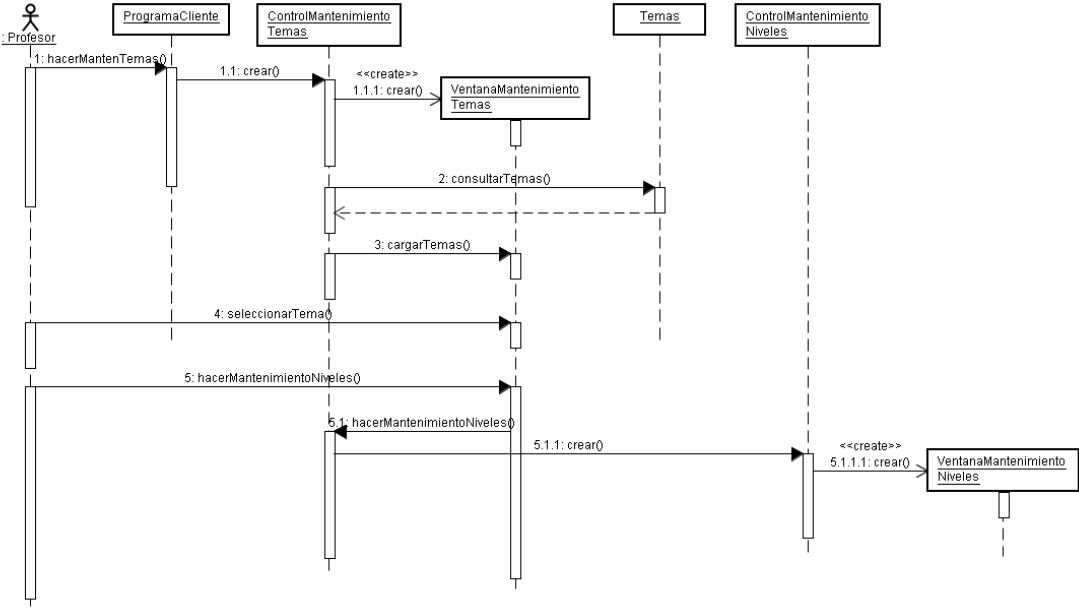
RF10 – Modificación de temas



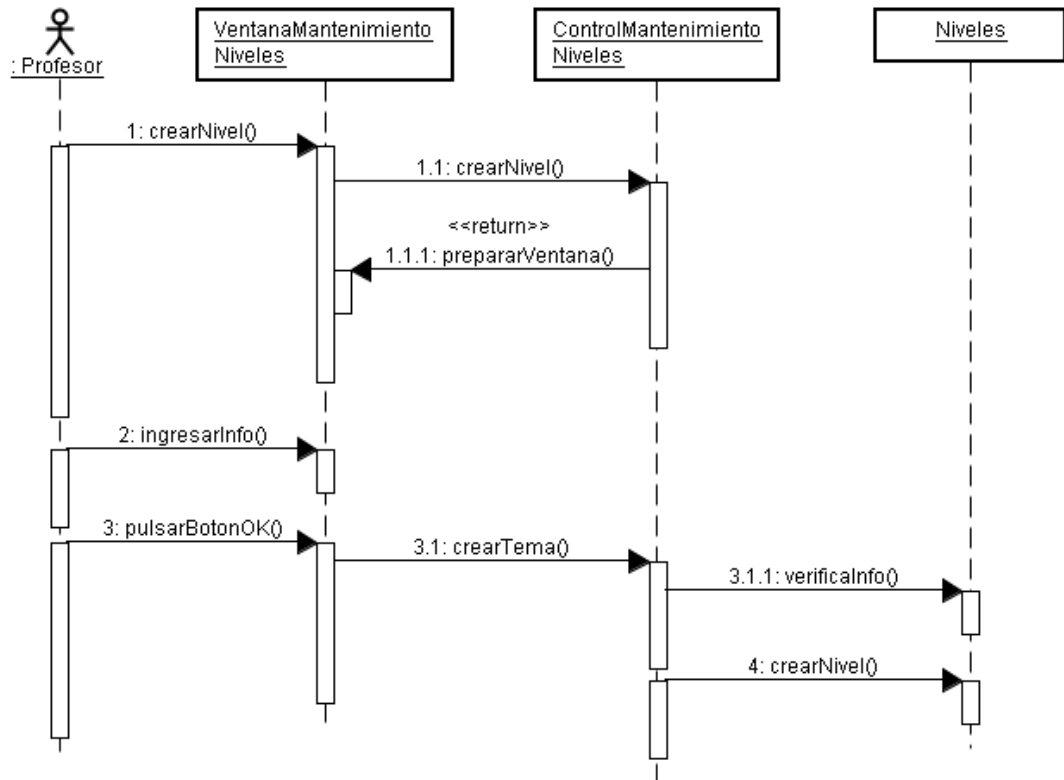
RF11 – Eliminación de temas



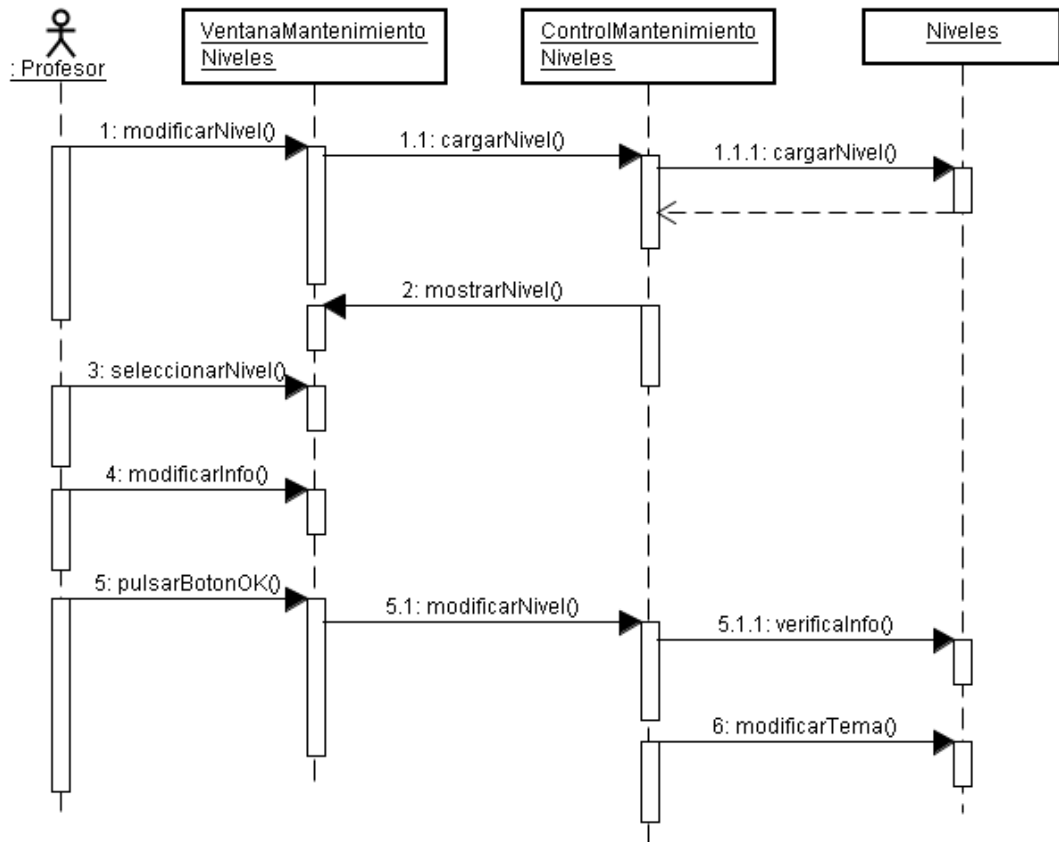
RF12 – Mantenimiento de niveles



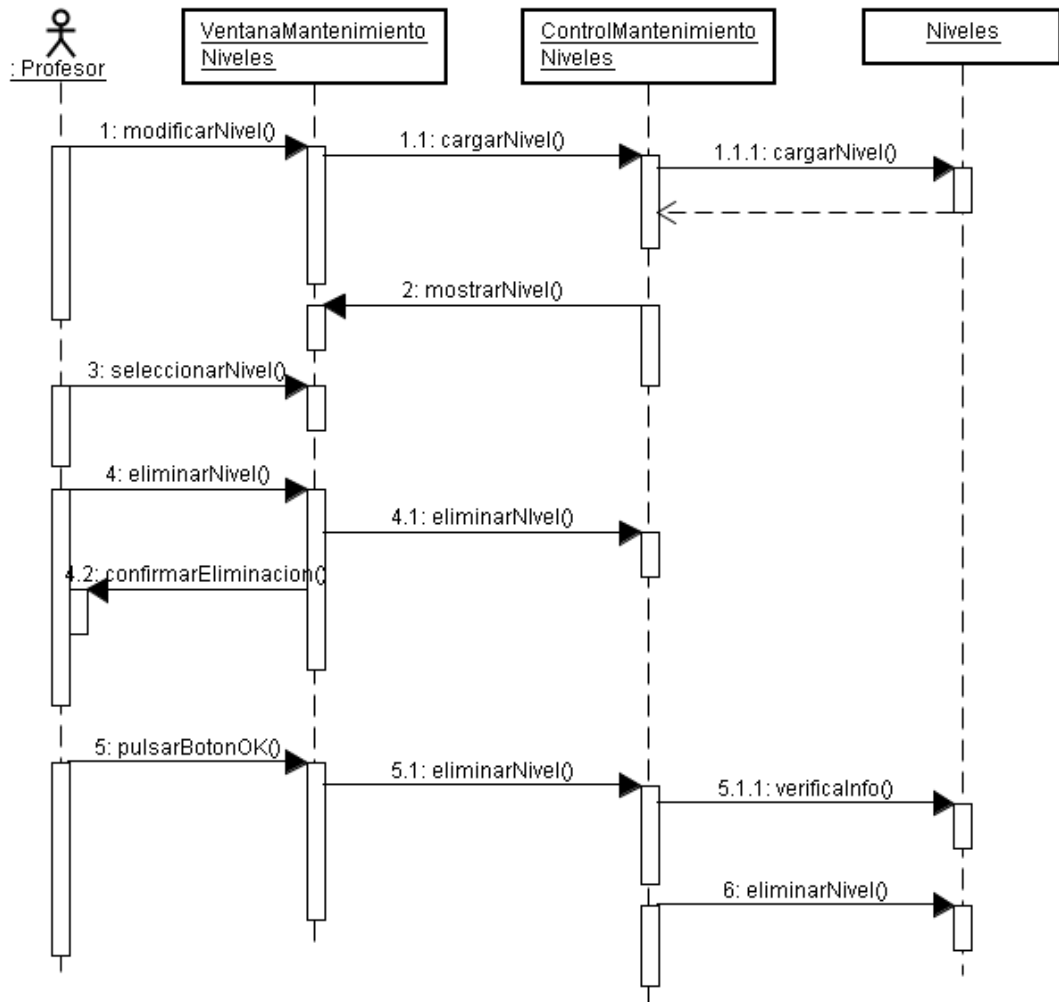
RF13 – Creación de niveles de dificultad



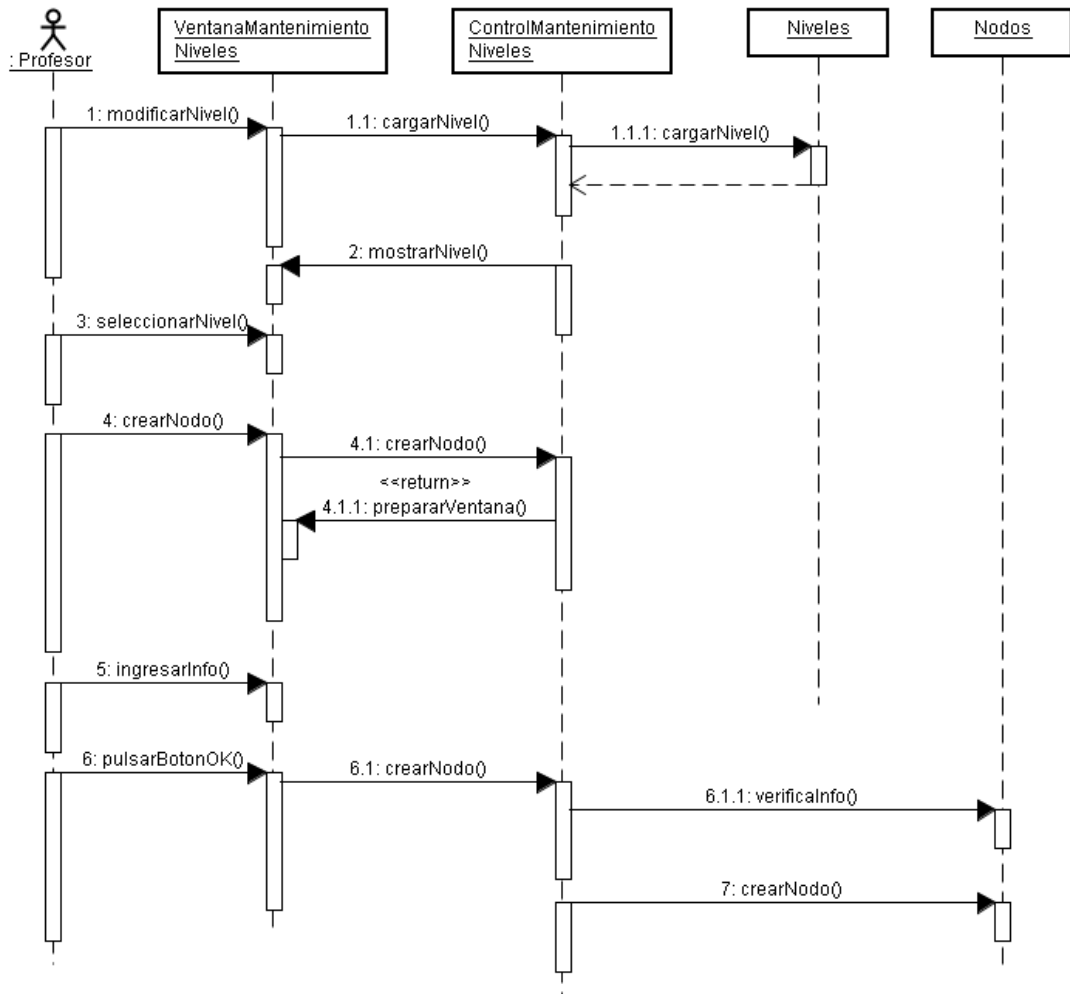
RF14 – Modificación de niveles de dificultad



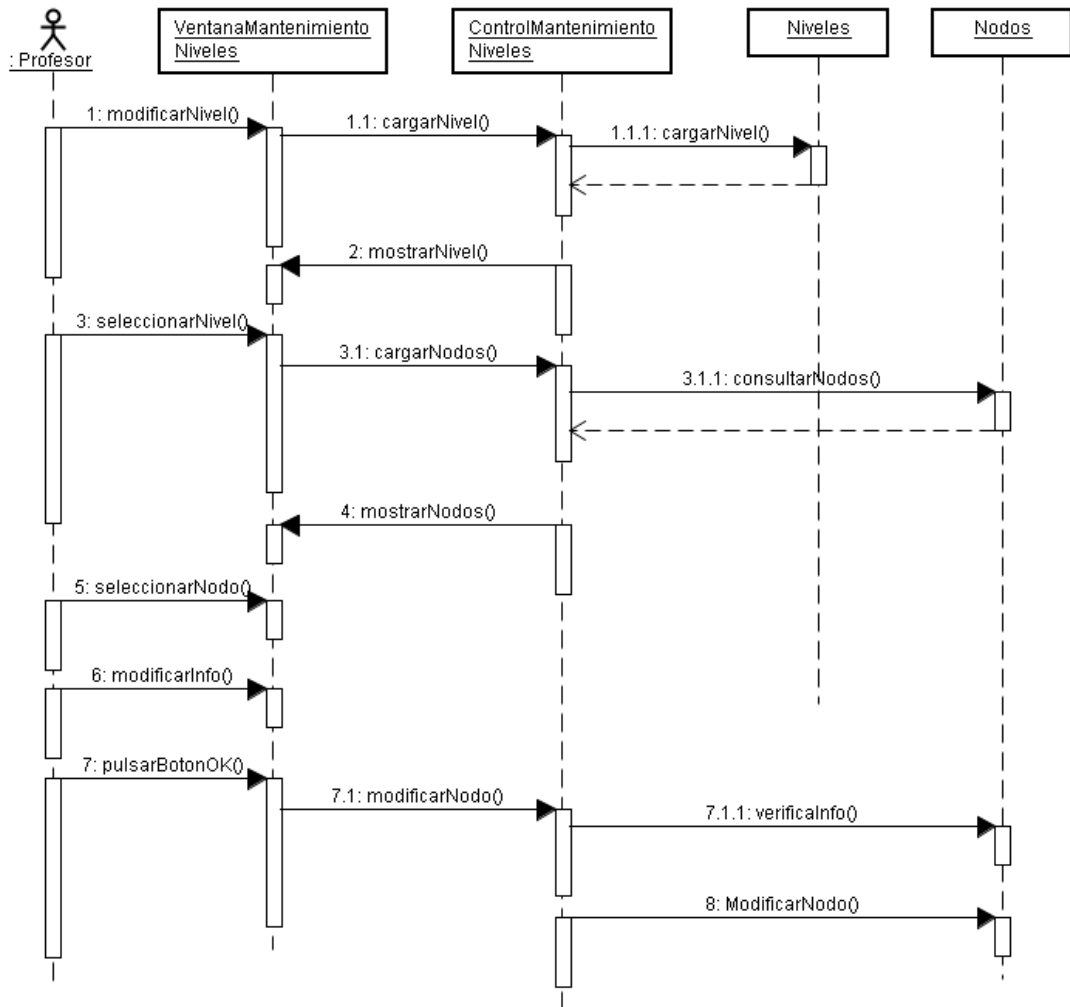
RF15 – Eliminación de niveles de dificultad



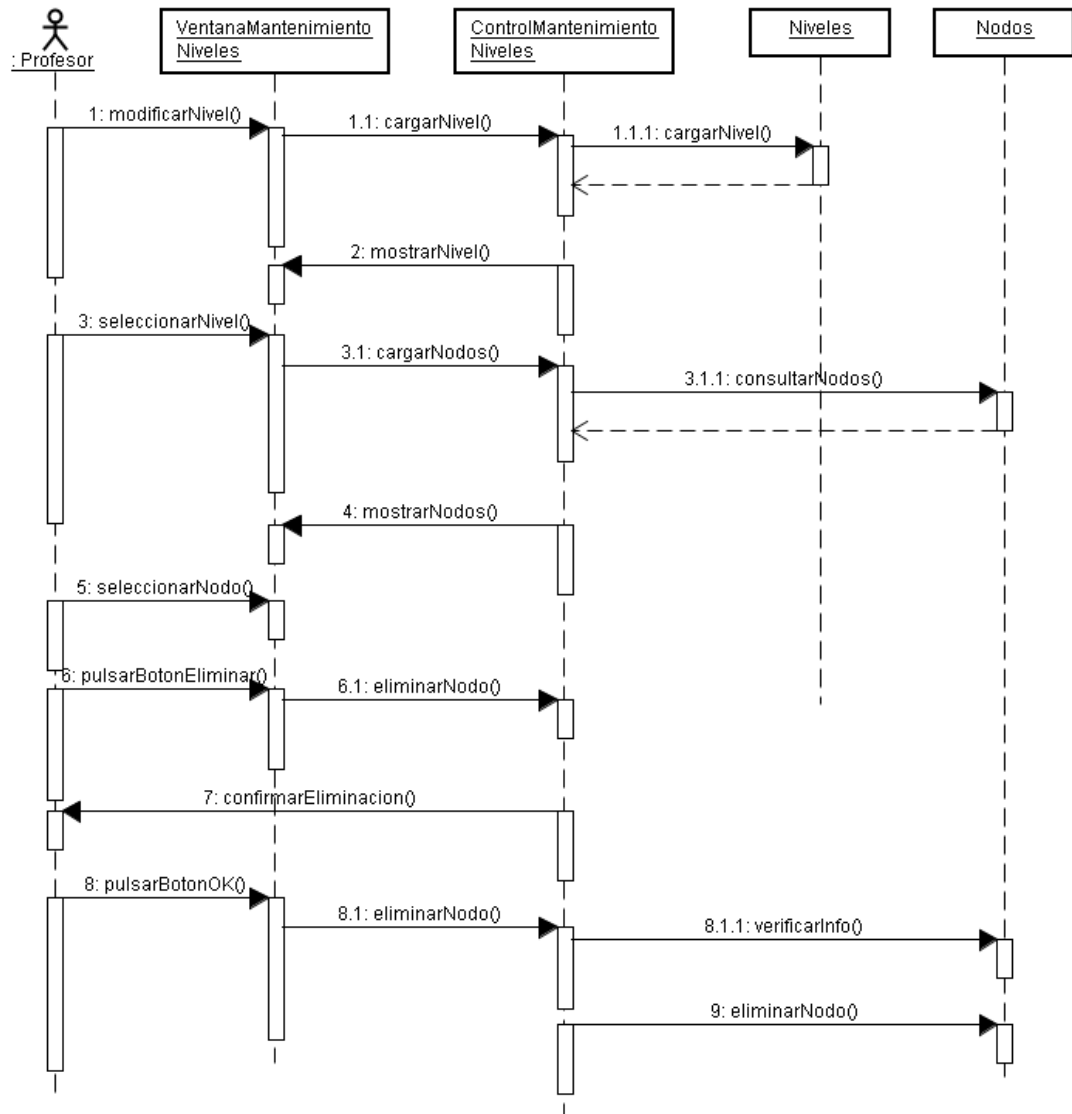
RF16 – Creación de nodos



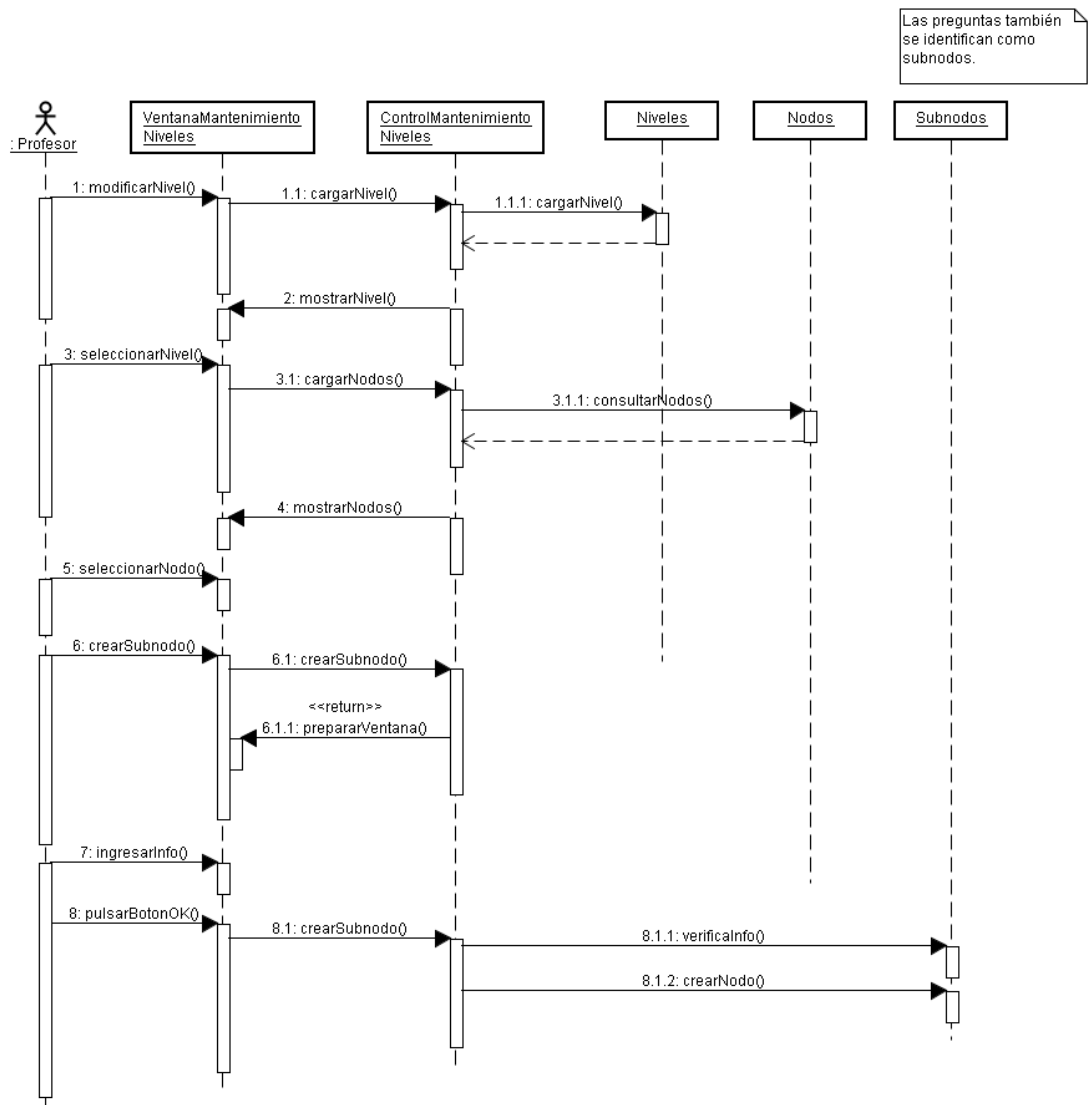
RF17 – Modificación de nodos



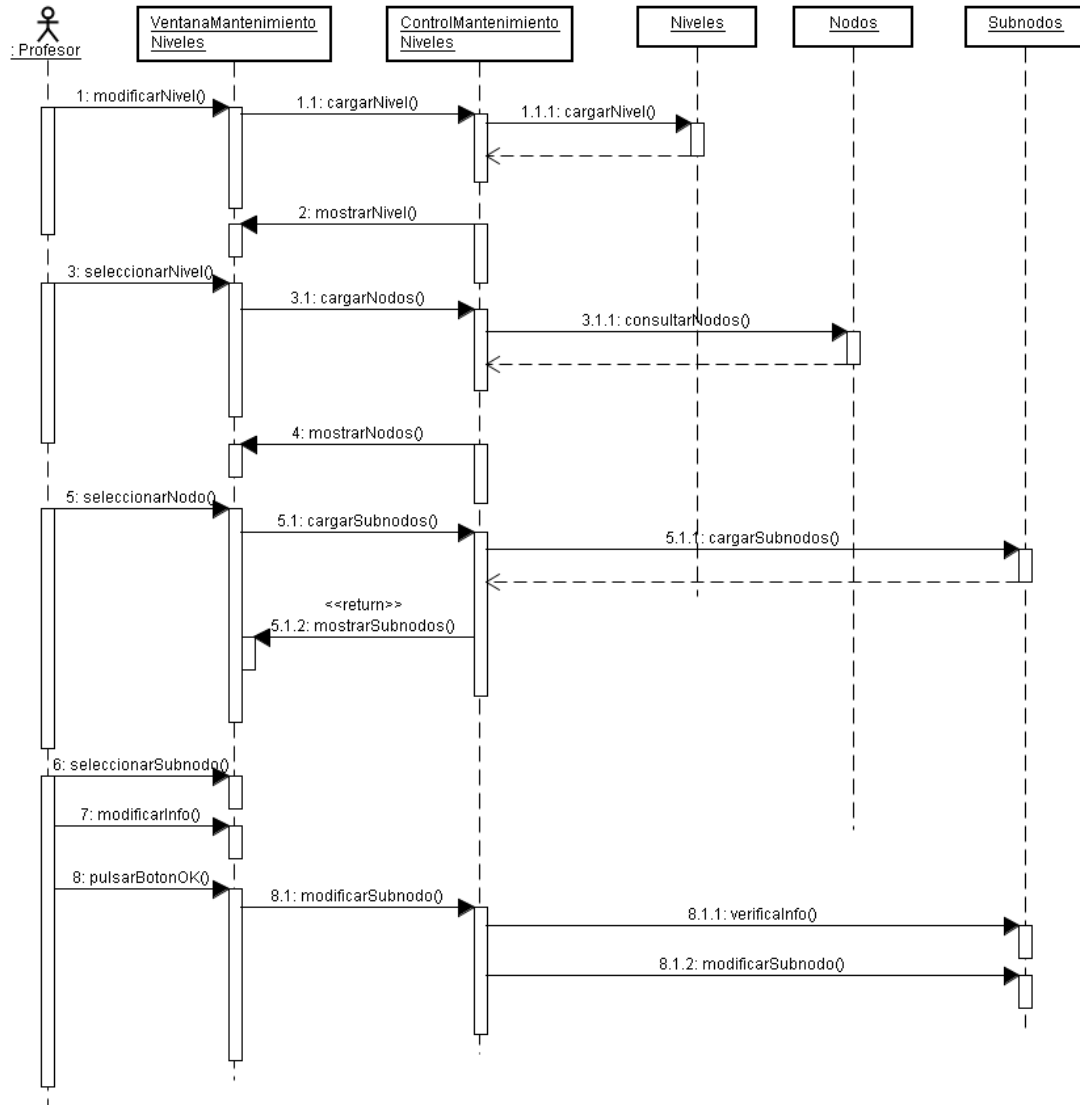
RF18 – Eliminación de nodos



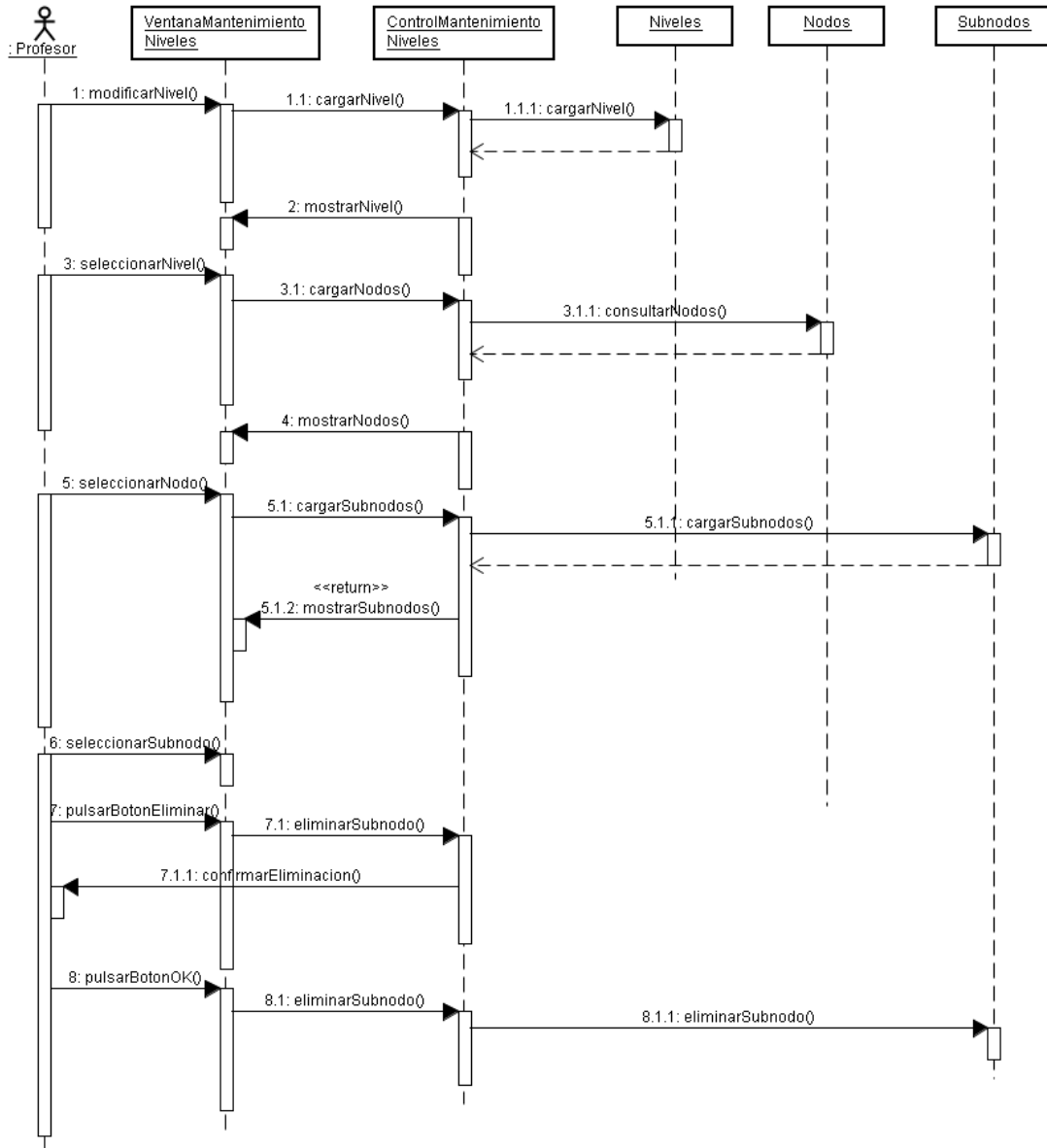
RF19 – Creación de preguntas



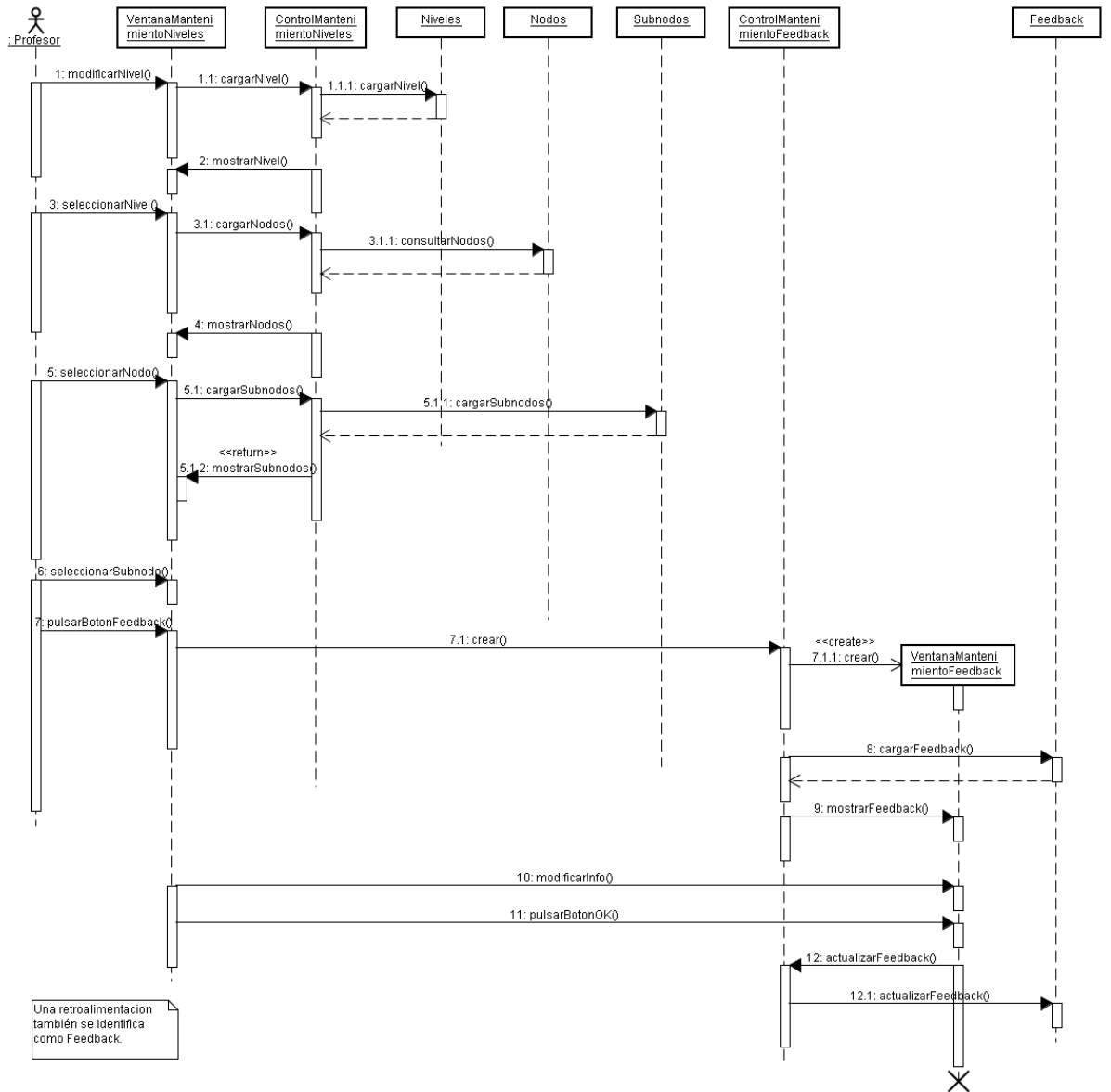
RF20 – Modificación de preguntas



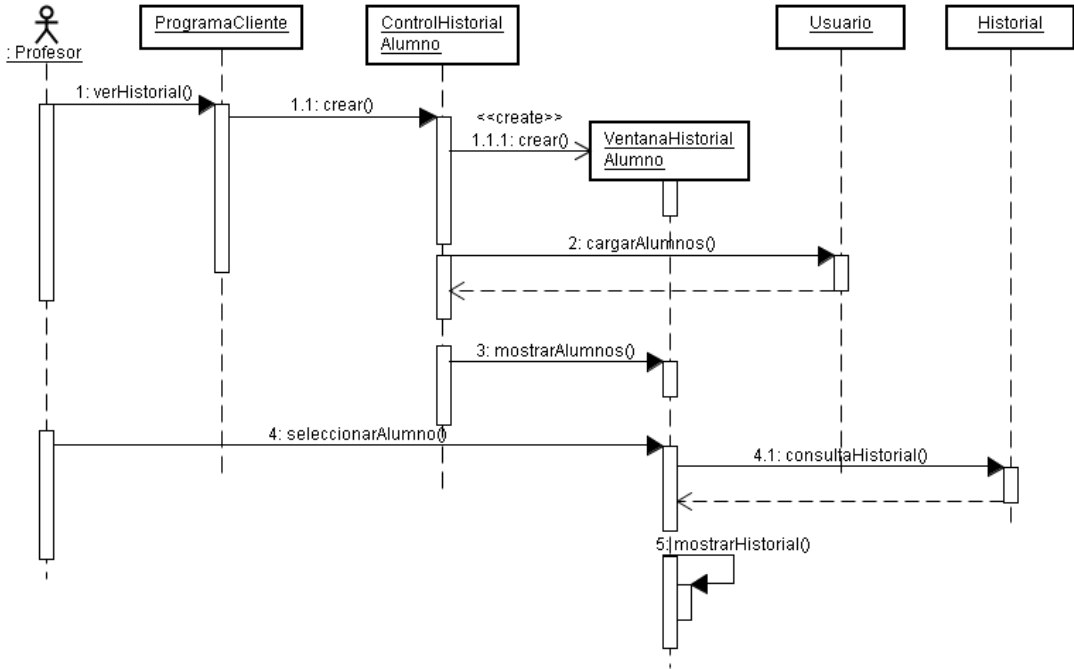
RF21 – Eliminación de preguntas



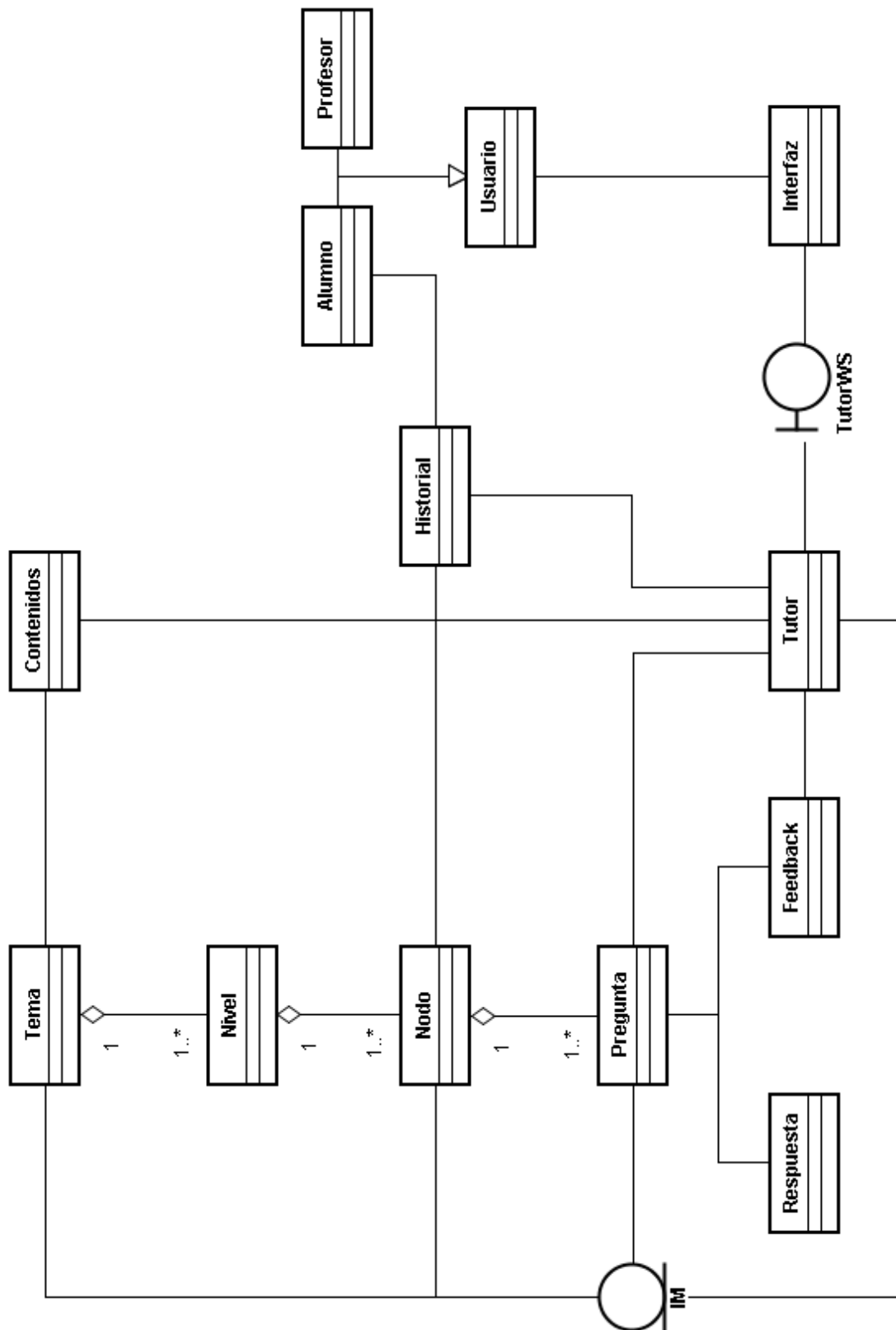
RF22 – Mantenimiento de retroalimentaciones



RF23 – Ver historial de un alumno

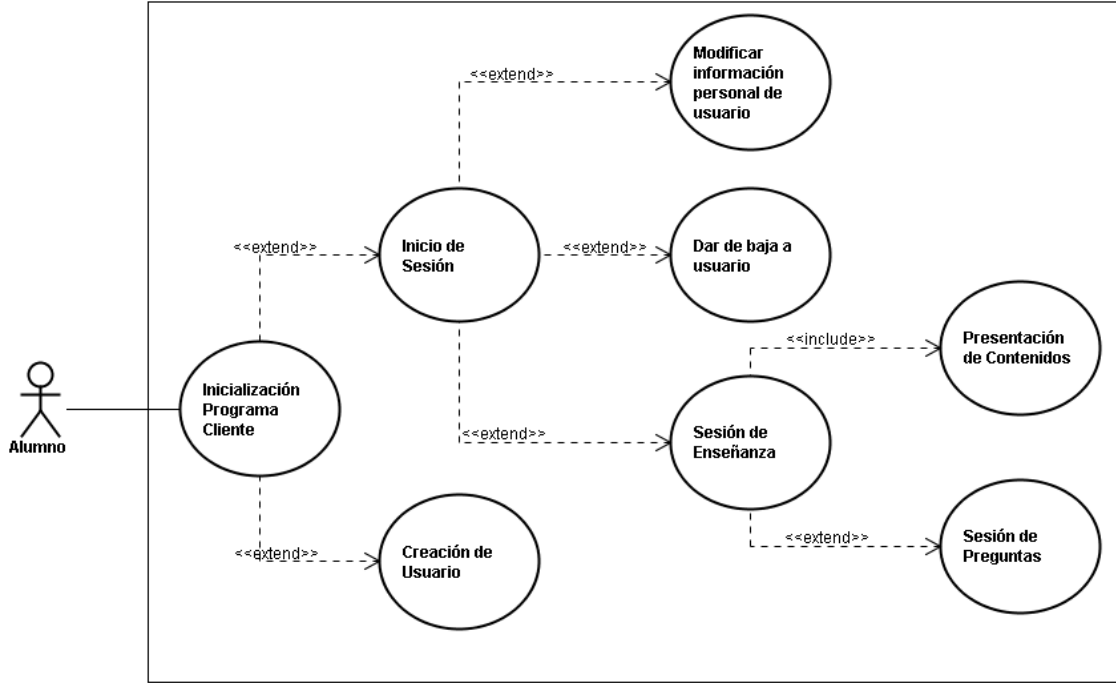


2.2. Diagrama de clases

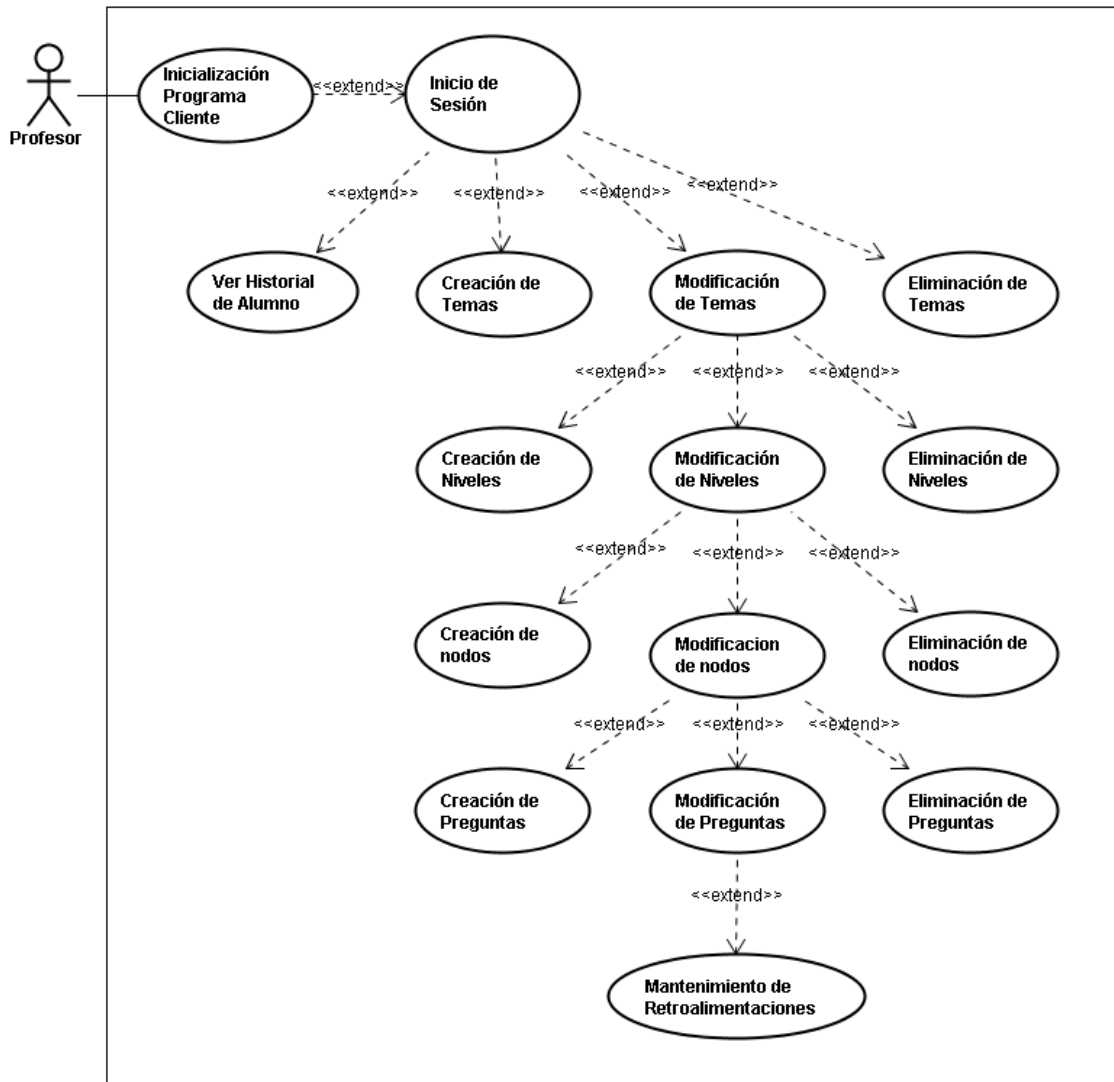


2.3. Diagramas de casos de uso

2.3.1. Casos de uso de alumno

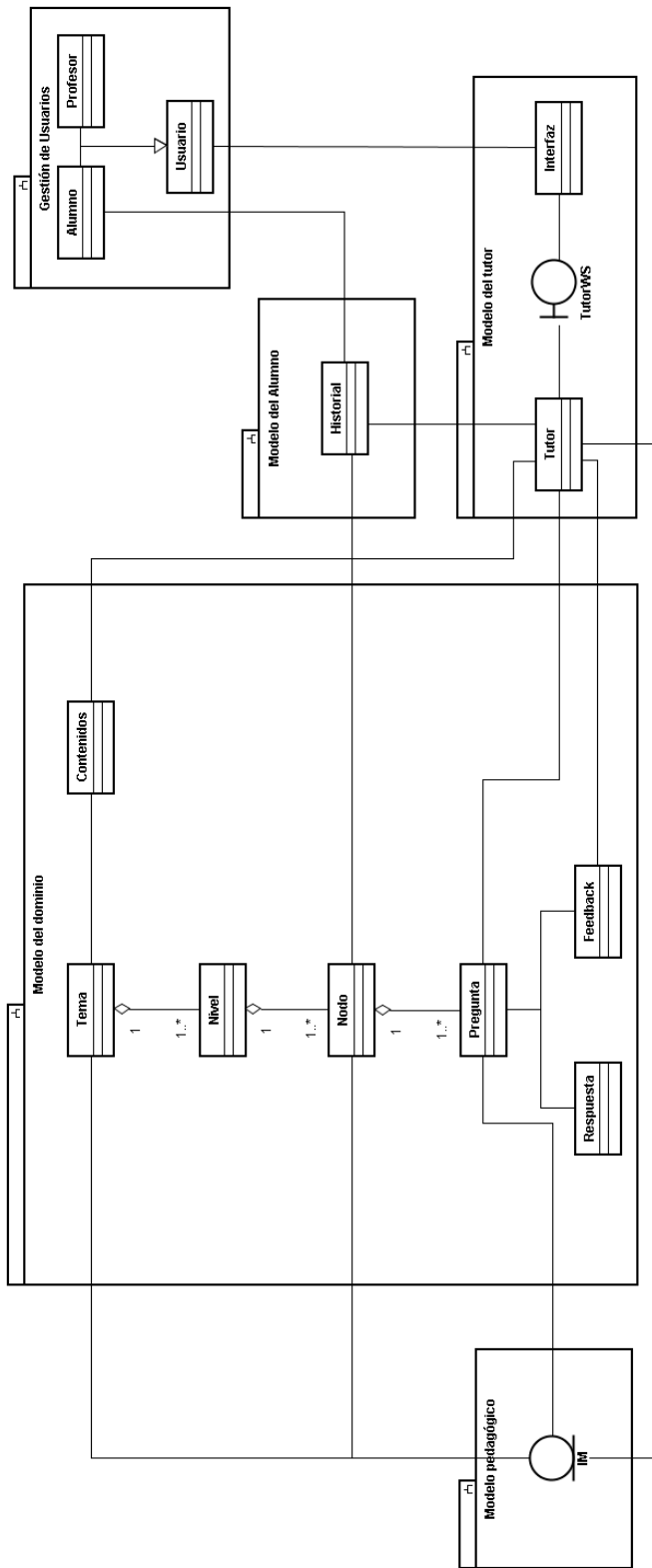


2.3.2. Casos de uso de profesor



3. Diseño del sistema

3.1. Descomposición en subsistemas



3.2. Descripción de subsistemas

3.2.1. Gestión de usuarios

Este subsistema es responsable de la administración de usuarios del sistema. En sus tareas está la creación, modificación y dada de baja de usuarios del sistema.

3.2.2. Modelo del dominio

Este subsistema representa la estructura jerarquizada del conocimiento disponible para la enseñanza. Es responsable del mantenimiento de temas de enseñanza, niveles de dificultad, nodos, preguntas, respuestas y retroalimentaciones (feedback) y del mantenimiento de relaciones entre estas estructuras para procurar la integridad de la organización de conocimiento.

3.2.3. Modelo del tutor

Le concierne la activación del servicio Web que atiende las solicitudes de usuarios y sus interfaces. Además es quién transmite las preguntas y respuestas entre usuario (alumno) y el componente tomador de decisiones del módulo pedagógico. Es responsable de mantener el historial de cada alumno registrado en el sistema.

3.2.4. Modelo pedagógico

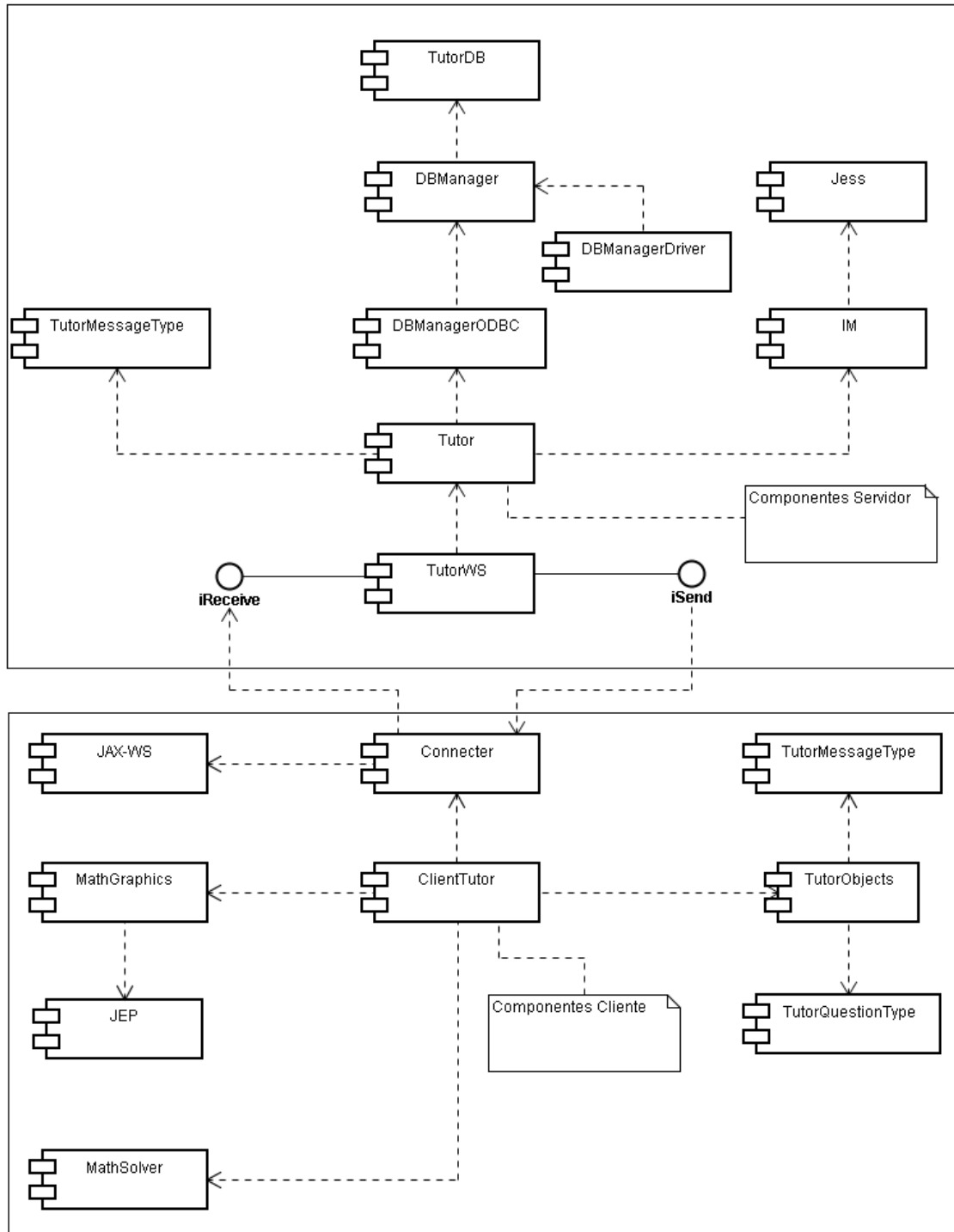
Este componente del sistema es responsable de la evaluación de las preguntas formuladas al alumno y de tomar decisiones sobre el nodo siguiente que debe visitar según sea la eficiencia de sus respuestas dadas.

3.2.5. Modelo del alumno

Es el garante del historial de seguimiento del alumno. Debe estar disponible para consultarse cada vez que se requiera, ya sea por el componente pedagógico para la toma de decisiones o de un usuario administrativo (profesor) para el análisis de un alumno en particular.

3.3. Correspondencia entre hardware y software

3.3.1. Descripción de componentes



TutorWS

Servicio Web que recibe las peticiones de los clientes y ofrece las funciones del componente tutor. Es la puerta de enlace entre el sistema tutor y la interfaz de usuario. Requiere de los componentes *Glassfish V2* y *Tutor*. Implementado en el proyecto.

GlassFish V2

Servidor de aplicaciones basado en Java Enterprise Edition, plataforma para la instalación del servicio Web *TutorWS*. Componente externo, *open source* bajo licenciamiento dual CDDL (Common Development and Distribution License) y GNU GPL con Classpath Exception (Licencia Pública General de GNU).

Tutor

Componente responsable de la preparación de las interfaces gráficas para la presentación de contenidos y la sesión de preguntas. Es función suya transmitir las respuestas del alumno al componente tomador de decisiones y almacenar en la base de datos el historial del alumno, conforme se vayan evaluando sus respuestas. Requiere de los componentes *IM* y *DBManagerODBC*. Es empleado por el componente *TutorWS*. Implementado en el proyecto.

TutorMessageType

Componente que guarda los mensajes de error entre las operaciones del tutor y la base de datos. Es empleado por el componente *Tutor*. Implementado en el proyecto.

DBManager

Componente responsable de realizar lecturas y escrituras en la base de datos. Es requerido por *DBManagerODBC* y *DBManagerDriver*. Implementado en el proyecto.

DBManagerODBC

Componente derivado de *DBManager*, responsable de establecer una conexión mediante el estándar de acceso a bases de datos ODBC. Es empleado por *Tutor*. Implementado en el proyecto.

DBManagerDriver

Componente derivado de *DBManager*, responsable de establecer una conexión utilizando el driver de comunicación nativo de Java con bases de datos (JDBC). No es empleado por *Tutor*, pero puede suplir al componente *DBManagerODBC* en caso de que se desee modificar el método de conexión a la base de datos. Implementado en el proyecto.

TutorDB

Base de datos del sistema. Soportada por un motor de bases de datos MySQL. Es requerida por el componente *DBManagerODBC* quien es el responsable de gestionarla. Implementada en el proyecto.

IM

Componente tomador de decisiones. Responsable de las evaluaciones a las respuestas y de indicar el siguiente nodo de comprobación de conocimiento por alumno. Requiere del componente *Jess*. Es empleado por el componente *Tutor*. Implementado en el proyecto.

Jess

Máquina de Inferencia de reglas para Java, basada en CLIPS. Provee una serie de funciones (API) para programación basada en reglas. Es empleado por el componente *IM*. Componente externo. Comercial. Disponible sin costo para uso académico.

ClientTutor

Es el componente que reside en la máquina que opera el usuario, mediante el cual interactúa con el sistema tutor utilizando una interfaz gráfica de usuario. Se descarga a través de un navegador Web. Es preparada por el componente *Tutor* a través del servicio Web *TutorWS*. Requiere de los componentes *Connecter*, *TutorObjects*, *MathGraphics* y *MathSolver*. Implementado en el proyecto.

Connecter

Componente que establece comunicación entre la interfaz gráfica de usuario, representada en el componente *ClientTutor*, y el sistema tutor. Requiere del conjunto de librerías *JAX-WS*. Es utilizado por el componente *ClientTutor*. Implementado en el proyecto.

JAX-WS

Conjunto de librerías que permiten la comunicación con un servicio Web. Es empleado por el componente *Connecter*. Componentes externos, *open source* bajo licenciamiento dual CDDL (Common Development and Distribution License) y GNU GPL con Classpath Exception (Licencia Pública General de GNU).

TutorObjects

Componente que encapsula los mensajes de error de la librería *TutorMessageType* y los tipos de preguntas que puede lanzar el tutor almacenadas en la librería *TutorQuestionType*. Es requerida por el componente *ClientTutor*. Implementado en el proyecto.

TutorQuestionType

Componente que guarda los tipos de preguntas que pueden ser lanzadas por el *Tutor* al usuario. Es empleado por el componente *ClientTutor*. Implementado en el proyecto.

MathGraphics

Componente responsable de la graficación de funciones matemáticas. Es utilizado por el componente *ClientTutor*. Requiere del componente *JEP*. Implementado en el proyecto.

JEP

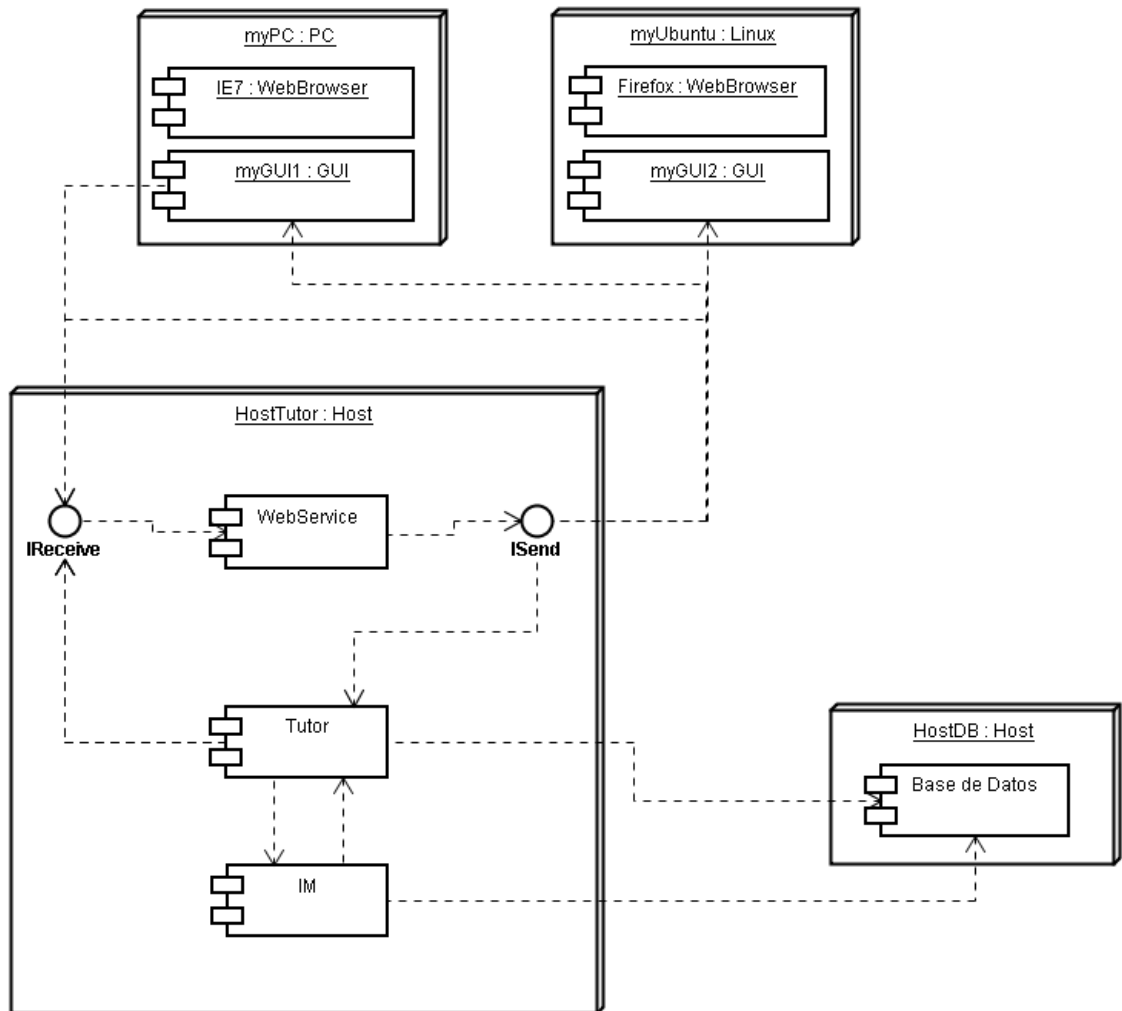
Componente utilizado para analizar e interpretar funciones matemáticas. Es utilizado por el componente *MathGraphics* para obtener los puntos para graficar una función matemática. Componente externo, *open source* bajo licenciamiento

GPL GNU (Licencia Pública General de GNU) hasta la versión 2.3.1, empleada en el proyecto.

MathSolver

Componente encargado de asistir al usuario en la solución de métodos numéricos para ingeniería. Es requerido por el componente *ClientTutor*. Implementado en el proyecto.

3.3.2. Despliegue físico de componentes



El sistema tutor inteligente, se ha implementado bajo una arquitectura cliente servidor en aras de que cualquier usuario registrado pueda acceder a una sesión de enseñanza sin importar su espacio geográfico sirviendo como base la Internet. Así lo indica el diagrama de despliegue mostrado sobre estas líneas.

La dimensión lógica de la implementación del proyecto se traduce en el desarrollo de componentes, constitución de clases e interfaces.

La dimensión física de este proyecto se representa en los procesadores o equipos que albergarán la dimensión lógica. Están descritos así:

- HostTutor: Este nodo es el responsable de empaquetar los servicios del sistema tutor, recepción y envío de preguntas, respuestas y retroalimentaciones; evaluación de respuestas y toma de decisiones.
- HostDB: Este nodo es el garante del almacenamiento de la base de datos del sistema, cuyo componente permite y participa en el buen funcionamiento de la ejecución del mismo puesto que proporciona la información necesaria para realizar todas las funciones del mismo. El componente de la base de datos está aislado en un nodo independiente, aunque debido a las bondades del motor de bases de datos empleado, MySQL, esta puede ser ubicada en el mismo nodo donde se almacenan los servicios del sistema.
- Máquina Cliente: Representa al nodo que accede a los servicios del sistema a través de una interfaz gráfica descargada usando un navegador Web.

Las máquinas HostTutor y HostDB, deben estar preferiblemente enlazadas dentro de una misma LAN (Red de Área Local). La comunicación entre una máquina cliente y el nodo que engloba los servicios del sistema se realiza usando la Web.

3.4. Administración de datos

Son identificados como información persistente dentro del flujo de eventos básico del sistema, los datos del alumno, y las estructuras jerarquizadas de los contenidos de aprendizaje. Se pretende el almacenamiento de estos datos en una base de datos relacional, en aras de permitir consultas, que puedan variar en grado de complejidad, según lo que se requiera.

El almacenaje de este volumen de información, se soportará en un sistema gestor de bases de datos MySQL, por las siguientes causas:

- Rendimiento: Es una de las características inherentes de este motor. Al ser un proyecto cuyo tránsito de información fluye a través de la Web, MySQL ofrece esta propiedad, obviando control y/o auditoría, facilitando la rapidez de las transacciones y ofreciendo mejores tiempos de respuesta a los componentes del sistema que acceden a la información.
- Integridad: Al ofrecer la posibilidad de crear relaciones fuertes (constraints) entre estructuras de datos, se procura que al afectar una de ellas afecte o no a sus descendientes.
- Portabilidad: MySQL ofrece independencia de plataforma (Windows, Linux, Solaris, etc.) en cuanto a instalación y administración. También es extensible a los estándares de conectividad de bases de datos JDBC (Java) y ODBC (Windows), los más utilizados en sistemas de información.
- Acoplamiento: MySQL ofrece fácil integración con el lenguaje Java, en el que se implementó el sistema tutor.
- Licenciamiento: La licencia *community* de la versión 5.0.45 de MySQL, sobre la que se cimentó este proyecto de software, es software libre y no representa costo alguno.

3.5. Control de acceso

La segregación de funcionalidades del sistema se realiza de acuerdo con el perfil de los usuarios que acceden al mismo. Este sistema es catalogado como multi-usuario. De este modo, y como lo indican los requerimientos del sistema, un usuario alumno sólo tendrá acceso a su información personal, visitar contenidos de enseñanza y realizar sesiones de preguntas y respuestas.

Las tareas administrativas son encargadas al usuario profesor quien es responsable del mantenimiento de los contenidos de enseñanza. Además, tiene permitido consultar el historial de un alumno.

Todos los usuarios, sin excepción deberán acceder al sistema, ingresando una dirección de correo electrónico válida y registrada y una contraseña de ocho caracteres mínimo sin espacios.

3.6. Flujo de control

El sistema admite la concurrencia de usuarios, pues su diseño, abierto y basado en Web hace de esta cualidad una característica inherente. El flujo de control se realiza a través de hilos o conexiones, que corresponden a eventos distintos posibilitando la múltiple interacción de usuarios en un mismo instante de tiempo.

3.7. Condiciones de frontera

Se puede acceder al sistema descargando una aplicación cliente a través de una página Web. No obstante, para acceder a los servicios que el sistema ofrece, es

necesario iniciar sesión para lo que se ha implementado un servicio de inicio de sesión para que los usuarios registrados puedan ingresar mediante la digitalización de una dirección de correo electrónico y su contraseña.

Las condiciones de funcionamiento están dadas siempre y cuando se respeten los siguientes aspectos:

- Configuración de la base de datos del sistema en una máquina que ofrezca conectividad ODBC.
- Óptima comunicación entre la máquina en la que reside la base de datos y la máquina en la que residen los servicios del sistema.
- Conectividad a Internet de la máquina donde se ejecute el programa cliente.

El manejo de los datos excepcionales, se realiza desplegando mensaje significativos en tiempo de ejecución ante el usuario, a modo de detectar y corregir a tiempo las fallas producidas en el sistema.

El apagado del sistema se realiza finalizando sesión a través de una opción habilitada para ello en el menú del programa cliente.

4. Modelado de datos

4.1. Estructura de la base de datos

4.1.1. Estructuras de parametrización

Tabla	Params		
Descripción	Definición de Parámetros		
Campo	Tipo	Descripción	
pr_id	INTEGER	Identificación de parámetro.	
pr_name	VARCHAR (45)	Descripción de parámetro.	
pr_value	VARCHAR (45)	Valor de parámetro.	
Índice	Tipo	Campos	Descripción
PRIMARY	PRIMARY	pr_id	Identificador único.

4.1.2. Estructuras asociadas a información de usuario

Tabla	Usuario		
Descripción	Información personal de usuario		
Campo	Tipo	Descripción	
us_id	INTEGER	Identificador único de usuario.	
us_email	VARCHAR (30)	Dirección de correo electrónico.	
us_nombre	VARCHAR (45)	Nombres de usuario.	
us_apellido	VARCHAR (45)	Apellidos de usuario.	
us_pswrd	VARCHAR (32)	Contraseña de usuario.	
us_sesion	INTEGER	Indicador de sesión iniciada.	
us_nodo	INTEGER	Último nodo visitado por usuario.	
us_estado	VARCHAR (1)	Indicador de vigencia (Activo / Inactivo).	
us_tipo	VARCHAR (1)	Tipo de usuario (Alumno / Profesor)	
Índice	Tipo	Campos	Descripción

us_idx1	UNIQUE	us_id, us_email	Identificador único de usuario.
---------	--------	-----------------	---------------------------------

4.1.3. Estructuras asociadas a organización de conocimiento

Tabla	Temas		
Descripción	Definición de temas de aprendizaje.		
Campo	Tipo	Descripción	
tm_tema	VARCHAR (3)	Identificador único de tema.	
tm_nombre	VARCHAR (45)	Descripción o nombre de tema.	
tm_prior	VARCHAR (3)	Identificador de tema anterior.	
tm_next	VARCHAR (3)	Identificador de tema siguiente.	
Índice	Tipo	Campos	Descripción
PRIMARY	PRIMARY	tm_tema	Identificador único de tema.

Tabla	Contenidos		
Descripción	Definición de contenidos asociados a un tema.		
Campo	Tipo	Descripción	
cn_id	INTEGER	Identificador único de contenido.	
cn_tema	VARCHAR (3)	Identificador de tema asociado.	
cn_url	VARCHAR (125)	Dirección URL del contenido.	
Índice	Tipo	Campos	Descripción
PRIMARY	PRIMARY	cn_id	Identificador único de contenido.
cn_idx1	UNIQUE	cn_id, cn_tema	Índice único. Identificador de contenido irrepitable por tema.

Tabla	Niveles		
Descripción	Definición de niveles de dificultad por tema		
Campo	Tipo	Descripción	
nv_nivel	INTEGER	Identificador de nivel.	
nv_tema	VARCHAR(3)	Identificador de tema asociado.	
nv_valmin	FLOAT	Valor mínimo de permisividad para eficiencia.	
nv_valmax	FLOAT	Valor máximo de permisividad para eficiencia.	
nv_permissiv	FLOAT	Tolerancia permitida para eficiencia.	
Índice	Tipo	Campos	Descripción
nv_idx1	UNIQUE	nv_nivel, nv_tema	Identificador único de registro.
nv_fk1	INDEX	nv_tema	Índice de llave foránea, para relación con tabla Temas.

Tabla	Nodos		
Descripción	Definición de grupos de preguntas por tema y nivel.		
Campo	Tipo	Descripción	
nd_id	INTEGER	Identificador de nodo.	
nd_tema	VARCHAR(3)	Identificador de tema asociado.	
nd_nivel	INTEGER	Identificador de nivel asociado.	
nd_subnivel	INTEGER	Dificultad de nodo.	
Índice	Tipo	Campos	Descripción
nd_idx1	UNIQUE	nd_id, nd_tema, nd_nivel	Identificador único de registro.
nd_fk1	INDEX	nd_nivel, nd_tema	Índice de llave foránea, para relación con tabla Niveles
nd_fk2	INDEX	nd_tema	Índice de llave foránea, para relación con tabla Temas.

Tabla	Subnodos		
-------	----------	--	--

Descripción	Definición de subnodos (preguntas) por nodo.		
Campo	Tipo	Descripción	
sb_id	INTEGER	Identificador de subnodo.	
sb_nod	INTEGER	Identificador de nodo asociado.	
sb_tipo	INTEGER	Tipo de subnodo.	
sb_problema	TEXT	Enunciado del problema.	
sb_opciones1	TEXT	Lista de opciones de selección.	
sb_opciones2	TEXT	Lista de opciones de selección (Aplica para preguntas de asociación).	
sb_conj1	VARCHAR (45)	Conjunto de opciones de selección.	
sb_conj2	VARCHAR (45)	Conjunto de opciones de selección (Aplica para preguntas de asociación).	
sb_resp	VARCHAR (45)	Conjunto de respuestas correctas.	
sb_peso	FLOAT	Ponderación de subnodo dentro de nodo. (Suma de ponderaciones de todos los subnodos de un nodo es igual a 100)	
sb_valmin	FLOAT	Valor mínimo de permisividad para respuesta.	
sb_valmax	FLOAT	Valor máximo de permisividad para respuesta.	
Índice	Tipo	Campos	Descripción
sb_idx1	UNIQUE	sb_id, sb_nod	Índice único.
sb_fk1	INDEX	sb_nod	Índice de llave foránea, para relación con tabla Nodos.

Tabla	Tipos		
Descripción	Definición de tipos de subnodos (preguntas).		
Campo	Tipo	Descripción	
tp_id	INTEGER	Identificador de tipo de subnodo.	
tp_descc	VARCHAR (10)	Descripción corta de tipo de subnodo.	
tp_descl	VARCHAR (45)	Descripción larga de tipo de subnodo.	
Índice	Tipo	Campos	Descripción
PRIMARY	PRIMARY	tp_id	Índice único.

Tabla	Feedback		
Descripción	Definición de retroalimentaciones por eficiencia alcanzada en subnodo (pregunta).		
Campo	Tipo	Descripción	
fb_subnodo	INTEGER	Identificador de subnodo asociado.	
fb_nivel	INTEGER	Nivel alcanzado en pregunta según respuesta.	
fb_feedback	TEXT	Retroalimentación según nivel.	
Índice	Tipo	Campos	Descripción
fb_idx1	UNIQUE	fb_subnodo, fb_nivel	Índice único.

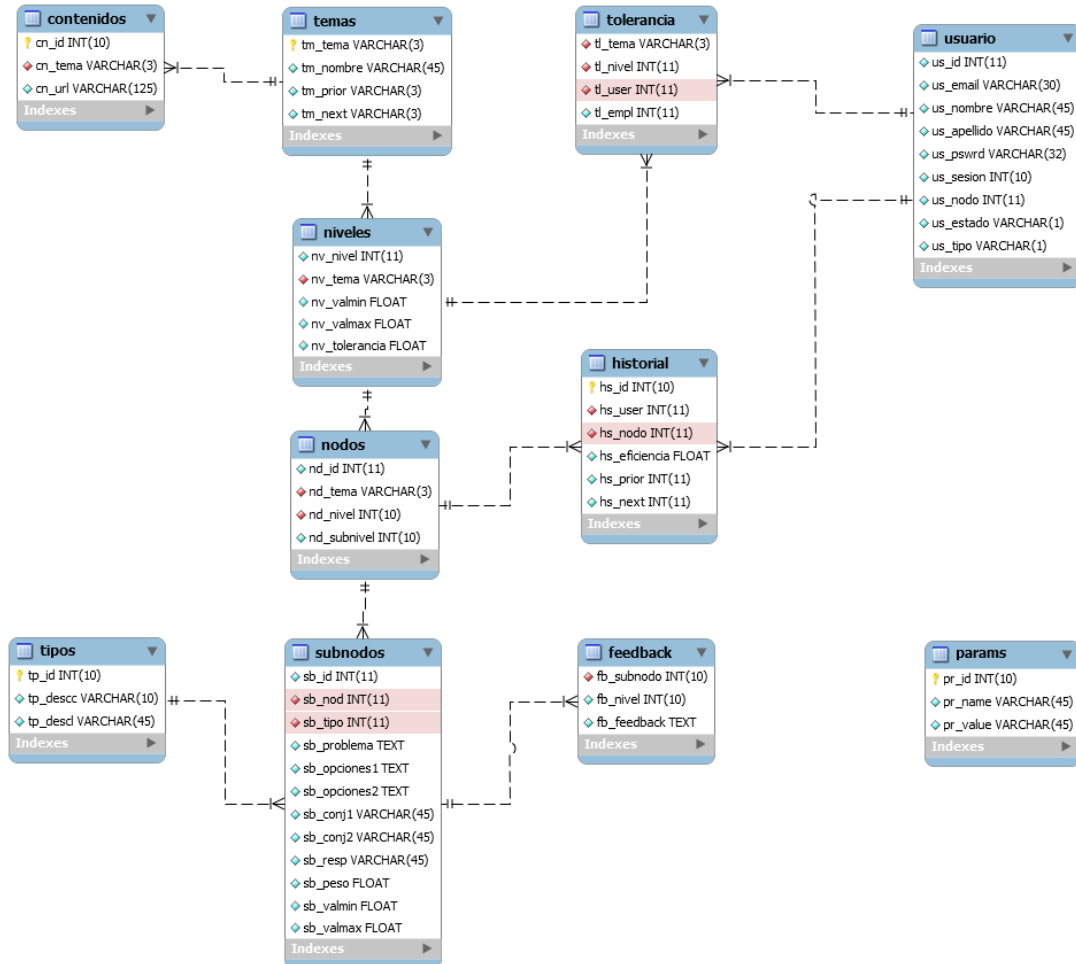
4.1.4. Estructuras que entrelazan información de usuario y organización de conocimiento

Tabla	Historial		
Descripción	Historial de consecución de conocimiento por usuario.		
Campo	Tipo	Descripción	
hs_id	INTEGER	Identificación de registro en historial.	
hs_user	INTEGER	Identificación de usuario.	
hs_nodo	INTEGER	Identificación de nodo actual.	
hs_eficiencia	FLOAT	Eficiencia para nodo actual.	
hs_prior	INTEGER	Identificación de nodo anterior.	
hs_next	INTEGER	Identificación de nodo siguiente.	
Índice	Tipo	Campos	Descripción
PRIMARY	PRIMARY	hs_id	Identificador único de registro en historial.
hs_idx1	UNIQUE	hs_user, hs_nodo, hs_id	Identificador único de registro por usuario y nodo.

Tabla	Tolerancia		
Descripción	Definición de tolerancia por usuario, tema y nivel		
Campo	Tipo	Descripción	
tl_tema	VARCHAR (3)	Identificación de tema.	
tl_nivel	INTEGER	Identificación de nivel.	
tl_user	INTEGER	Identificación de usuario.	
tl_empl	INTEGER	Indicador de tolerancia empleada.	
Índice	Tipo	Campos	Descripción
tl_idx1	UNIQUE	tl_tema, tl_nivel, tl_user	Índice único de registro.
tl_fk1	INDEX	tl_nivel, tl_tema	Índice de llave foránea, para relación con tabla Niveles.
tl_fk2	INDEX	tl_user	Índice de llave foránea, para

relación con tabla Usuarios.

4.2. Modelo entidad-relación de la base de datos



5. Licenciamiento

5.1. GNU Lesser General Public License

Version 3, 29 June 2007

Copyright (C) 2007 Free Software Foundation, Inc. <<http://fsf.org/>>

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

This version of the GNU Lesser General Public License incorporates the terms and conditions of version 3 of the GNU General Public License, supplemented by the additional permissions listed below.

0. Additional Definitions.

As used herein, “this License” refers to version 3 of the GNU Lesser General Public License, and the “GNU GPL” refers to version 3 of the GNU General Public License.

“The Library” refers to a covered work governed by this License, other than an Application or a Combined Work as defined below.

An “Application” is any work that makes use of an interface provided by the Library, but which is not otherwise based on the Library. Defining a subclass of a class defined by the Library is deemed a mode of using an interface provided by the Library.

A “Combined Work” is a work produced by combining or linking an Application with the Library. The particular version of the Library with which the Combined Work was made is also called the “Linked Version”.

The “Minimal Corresponding Source” for a Combined Work means the Corresponding Source for the Combined Work, excluding any source code for portions of the Combined Work that, considered in isolation, are based on the Application, and not on the Linked Version.

The “Corresponding Application Code” for a Combined Work means the object code and/or source code for the Application, including any data and utility programs needed for reproducing the Combined Work from the Application, but excluding the System Libraries of the Combined Work.

1. Exception to Section 3 of the GNU GPL.

You may convey a covered work under sections 3 and 4 of this License without being bound by section 3 of the GNU GPL.

2. Conveying Modified Versions.

If you modify a copy of the Library, and, in your modifications, a facility refers to a function or data to be supplied by an Application that uses the facility (other than as an argument passed when the facility is invoked), then you may convey a copy of the modified version:

- * a) under this License, provided that you make a good faith effort to ensure that, in the event an Application does not supply the function or data, the facility still operates, and performs whatever part of its purpose remains meaningful, or

- * b) under the GNU GPL, with none of the additional permissions of this License applicable to that copy.

3. Object Code Incorporating Material from Library Header Files.

The object code form of an Application may incorporate material from a header file that is part of the Library. You may convey such object code under terms of your choice, provided that, if the incorporated material is not limited to numerical

parameters, data structure layouts and accessors, or small macros, inline functions and templates (ten or fewer lines in length), you do both of the following:

- * a) Give prominent notice with each copy of the object code that the Library is used in it and that the Library and its use are covered by this License.

- * b) Accompany the object code with a copy of the GNU GPL and this license document.

4. Combined Works.

You may convey a Combined Work under terms of your choice that, taken together, effectively do not restrict modification of the portions of the Library contained in the Combined Work and reverse engineering for debugging such modifications, if you also do each of the following:

- * a) Give prominent notice with each copy of the Combined Work that the Library is used in it and that the Library and its use are covered by this License.

- * b) Accompany the Combined Work with a copy of the GNU GPL and this license document.

- * c) For a Combined Work that displays copyright notices during execution, include the copyright notice for the Library among these notices, as well as a reference directing the user to the copies of the GNU GPL and this license document.

* d) Do one of the following:

0) Convey the Minimal Corresponding Source under the terms of this License, and the Corresponding Application Code in a form suitable for, and under terms that permit, the user to recombine or relink the Application with a modified version of the Linked Version to produce a modified Combined Work, in the manner specified by section 6 of the GNU GPL for conveying Corresponding Source.

1) Use a suitable shared library mechanism for linking with the Library. A suitable mechanism is one that (a) uses at run time a copy of the Library already present on the user's computer system, and (b) will operate properly with a modified version of the Library that is interface-compatible with the Linked Version.

* e) Provide Installation Information, but only if you would otherwise be required to provide such information under section 6 of the GNU GPL, and only to the extent that such information is necessary to install and execute a modified version of the Combined Work produced by recombining or relinking the Application with a modified version of the Linked Version. (If you use option 4d0, the Installation Information must accompany the Minimal Corresponding Source and Corresponding Application Code. If you use option 4d1, you must provide the Installation Information in the manner specified by section 6 of the GNU GPL for conveying Corresponding Source.)

5. Combined Libraries.

You may place library facilities that are a work based on the Library side by side in a single library together with other library facilities that are not Applications and are not covered by this License, and convey such a combined library under terms of your choice, if you do both of the following:

- * a) Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library facilities, conveyed under the terms of this License.

- * b) Give prominent notice with the combined library that part of it is a work based on the Library, and explaining where to find the accompanying uncombined form of the same work.

6. Revised Versions of the GNU Lesser General Public License.

The Free Software Foundation may publish revised and/or new versions of the GNU Lesser General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library as you received it specifies that a certain numbered version of the GNU Lesser General Public License “or any later version” applies to it, you have the option of following the terms and conditions either of that published version or of any later version published by the Free Software Foundation. If the Library as you received it does not specify a version number of the GNU Lesser General Public License, you may choose any version of the GNU Lesser General Public License ever published by the Free Software Foundation.

If the Library as you received it specifies that a proxy can decide whether future versions of the GNU Lesser General Public License shall apply, that proxy's public statement of acceptance of any version is permanent authorization for you to choose that version for the Library.

5.2. Licencia Pública General Reducida GNU

This is an unofficial translation of the GNU Lesser General Public License into spanish. It was not published by the Free Software Foundation, and does not legally state the distribution terms for software that uses the GNU LGPL—only the original English text of the GNU GPL does that. However, we hope that this translation will help spanish speakers understand the GNU GPL better.

Esta es una traducción no oficial de la Licencia Pública General Reducida GNU (GNU LGPL) al español. No fue publicada por la Fundación para el Software Libre, y no establece legalmente los términos de distribución para software que utiliza la GNU LGPL—sóamente el texto original en inglés de la GNU LGPL hace eso. De todas formas, esperamos que esta traducción ayude a los hispanohablantes a comprender mejor la GNU GPL.

Versión 3, 29 de junio 2007

Copyright (C) 2007 Free Software Foundation, Inc. <<http://fsf.org/>>

Se permite la copia y distribución de copias literales de esta licencia, pero no está permitido modificarla.

Esta versión de la Licencia Pública General Reducida GNU incorpora los términos y condiciones de la versión 3 de la Licencia Pública General GNU, extendida con los permisos adicionales que se listan a continuación.

0. Definiciones adicionales.

En adelante, "esta Licencia" se refiere a la versión 3 de la Licencia Pública General Reducida GNU, y "GNU GPL" a la versión 3 de la Licencia Pública General GNU.

"La Biblioteca" se refiere a un trabajo amparado por esta Licencia, que no sea una Aplicación o Proyecto Combinado según la definición que sigue.

Una "Aplicación" se refiere a cualquier trabajo que haga uso de la Biblioteca, pero sin estar basado en ella. Definir subclases a partir de clases de la Biblioteca se considera una forma de uso de la Biblioteca.

Un "Trabajo Combinado" es un trabajo realizado enlazando o combinando una Aplicación con la Biblioteca. La versión en particular de la Biblioteca con la cual se realizó el Trabajo Combinado se llama también "Versión Enlazada".

El "Código Fuente Mínimo" de un Trabajo Combinado se refiere al Código Fuente del Trabajo Combinado excluyendo cualquier parte del mismo que, considerada de forma aislada, esté basada en la Aplicación y no en la Versión Enlazada.

El "Código Correspondiente de la Aplicación" de un Trabajo Combinado se refiere al código objeto y/o al código fuente de la Aplicación, incluyendo cualquier dato o

programa auxiliar necesario para reproducir el Trabajo Combinado desde la Aplicación, sin incluir las Bibliotecas de Sistema implicadas en el Trabajo Combinado.

1. Excepción a la Cláusula 3 de la GNU GPL.

Usted puede distribuir un trabajo amparado bajo las condiciones de las cláusulas 3 y 4 de esta Licencia sin estar limitado por la cláusula 3 de la GNU GPL.

2. Distribución de versiones modificadas.

Si usted modifica una copia de la Biblioteca, y, en sus modificaciones se hace referencia desde alguna función a un servicio o datos suministrados por una Aplicación que utilice esa función (exceptuando los argumentos pasados al realizar la llamada a la función), entonces usted puede distribuir una copia de la versión modificada:

* a) bajo los términos y condiciones de esta Licencia, siempre que haga un esfuerzo razonable para asegurar que, en el caso de que la Aplicación no ofrezca la función o los datos, la utilidad siga funcionando y realice cualquier parte de sus propósitos de manera aceptable, o

* b) bajo los términos y condiciones de la Licencia General GNU (GNU GPL), sin que sea aplicable ningún permiso adicional de esta Licencia a esa copia.

3. Código Objeto que incorpora material de archivos de cabecera de la Biblioteca.

El código objeto de una Aplicación podrá incorporar contenido de un archivo de cabecera que forme parte de la Biblioteca. Usted podrá distribuir ese código objeto bajo los términos que desee, siempre y cuando, si el contenido incorporado no está limitado por parámetros numéricos, configuraciones de estructuras de datos y derivadas, o pequeñas macros, funciones incrustadas y plantillas (de diez o menos líneas de longitud), usted cumpla las dos siguientes condiciones:

* a) Avise de forma clara en cada copia del código objeto que ha utilizado la Biblioteca y que la Biblioteca y su uso están cubiertos por esta Licencia.

* b) Acompañe el código objeto de una copia de la Licencia GNU GPL y de este documento de licencia.

4. Trabajos combinados.

Usted podrá distribuir un Trabajo Combinado bajo los términos que desee, siempre que en conjunto, no haya efectos restrictivos sobre la modificación de

extractos de la Biblioteca que formen parte del Trabajo Combinado, ni sobre tareas de ingeniería inversa cuyo objetivo sea depurar tales modificaciones, si cumple además lo siguiente:

- * a) Avisa de forma clara en cada copia del código objeto que ha utilizado la Biblioteca y que la Biblioteca y su uso están cubiertos por esta Licencia.

- * b) Acompaña el código objeto de una copia de la Licencia GNU GPL y de este documento de licencia.

- * c) Incluye, en Trabajos Combinados que muestren avisos de Copyright durante su ejecución, el aviso adicional de copyright para la Biblioteca, así como referencias a copias de la Licencia GNU GPL y a este documento de licencia.

- * d) Cumpla una de la siguientes condiciones:

- 0) Distribuye el Código Fuente Mínimo bajo los términos de esta Licencia, y el Código de la Aplicación Correspondiente de forma apropiada, y bajo términos que permitan al usuario recombinar o volver a enlazar la aplicación con una versión modificada de la Versión Enlazada que permita obtener un Trabajo Combinado modificado, de la forma descrita en la cláusula 6 de la Licencia GNU GPL acerca de la distribución de la Fuente Correspondiente.

- 1) Utiliza un mecanismo de bibliotecas compartidas apropiado para enlazarse a la Biblioteca. Un mecanismo apropiado es aquel que (a) utiliza en tiempo de ejecución una copia de la Biblioteca existente en el ordenador del usuario, y (b) funcionará correctamente con una versión modificada de la Biblioteca que tenga una interfaz compatible con la Versión Enlazada.

* e) Proporciona Instrucciones de Instalación, pero sólo si no tuviese la obligación de proporcionar esas instrucciones de acuerdo con la cláusula 6 de la Licencia GNU GPL, y limitandose a las instrucciones necesarias para instalar y ejecutar una versión modificada del Trabajo Combinado producido al recombinar o enlazar de nuevo la Aplicación con una versión modificada de la Versión Enlazada. (Si elige la opción 4-d-0, las Instrucciones de Instalación deben adjuntar el Código Fuente Mínimo y el Código Correspondiente de la Aplicación. Si elige la opción 4-d-1, debe proporcionar las Instrucciones de Instalación de la forma descrita por la cláusula 6 de la Licencia GNU GPL acerca de la distribución de la Fuente Correspondiente.)

5. Bibliotecas Combinadas.

Usted podrá agrupar en una única biblioteca aquellas funciones que estén basadas en la Biblioteca, junto a otras funciones que no sean Aplicaciones y no estén cubiertas por esta Licencia, así como distribuir esa biblioteca combinada bajo los términos de su elección, siempre que cumpla las dos condiciones que siguen:

* a) Acompañe la biblioteca combinada con una copia del mismo trabajo que se base en la Biblioteca, sin combinar con ninguna otra función de biblioteca, y distribuida bajo los términos y condiciones de esta Licencia.

* b) Avise de forma clara en la biblioteca combinada que parte de la misma está basada en la Biblioteca, explicando dónde encontrar ese mismo trabajo sin combinar.

6. Versiones Modificadas de la Licencia Pública General Reducida GNU.

La Fundación para el Software Libre podrá publicar revisiones y/o nuevas versiones de la Licencia Pública General Reducida GNU de vez en cuando. Esas versiones serán similares en espíritu a la versión actual, pero podrán diferir en algunos detalles para afrontar nuevos problemas o situaciones.

A cada versión se le da un número distintivo. Si el Programa especifica que le es aplicable cierto número de versión de la Licencia Pública General Reducida o "cualquier versión posterior", usted tendrá la posibilidad de adoptar los términos y condiciones de la versión indicada o de cualquier otra versión posterior publicada por la Fundación para el Software Libre. Si el Programa no especifica un número de versión de la Licencia Pública General Reducida, usted podrá elegir cualquier versión que haya sido publicada por la Fundación para el Software Libre.

Si la Biblioteca, tal y como la recibe, especifica que un apoderado/representante puede decidir qué versiones de la Licencia Pública General Reducida pueden aplicarse en el futuro, la declaración pública de aceptación que el

apoderado/representante haga de una versión le autoriza a usted con carácter permanente a elegir esa versión para el Programa.