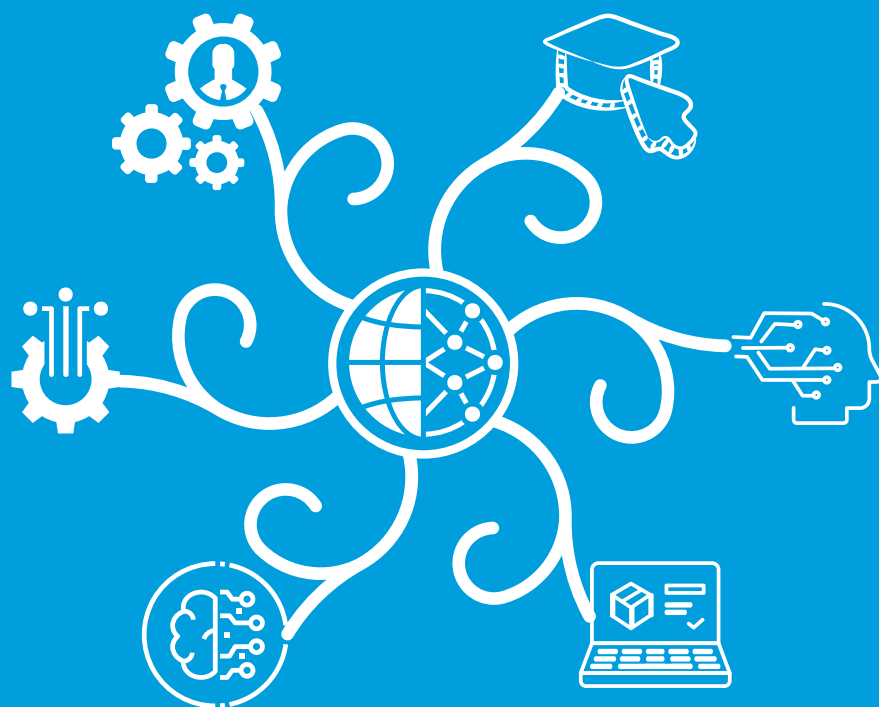


Sistemas inteligentes aplicados a la educación



Ramón Zatarain Cabada
María Lucía Barrón Estrada
Karina Mariela Figueroa Mora

Sistemas inteligentes aplicados a la educación

RAMÓN ZATARAIN CABADA
MARÍA LUCÍA BARRÓN ESTRADA
KARINA MARIELA FIGUEROA MORA



**COMUNICACIÓN
CIENTÍFICA**

Ediciones Comunicación Científica se especializa en la publicación de conocimiento científico de calidad en español e inglés en soporte de libro impreso y digital en las áreas de humanidades, ciencias sociales y ciencias exactas. Guía su criterio de publicación cumpliendo con las prácticas internacionales: dictaminación de pares ciegos externos, autenticación antiplagio, comités y ética editorial, acceso abierto, métricas, campaña de promoción, distribución impresa y digital, transparencia editorial e indexación internacional.

Cada libro de la Colección Ciencia e Investigación es evaluado para su publicación mediante el sistema de dictaminación de pares externos y autenticación antiplagio. Invitamos a ver el proceso de dictaminación transparentado, así como la consulta del libro en Acceso Abierto.



www.comunicacion-cientifica.com

[DOI.ORG/10.52501/cc.096](https://doi.org/10.52501/cc.096)




**COMUNICACIÓN
CIENTÍFICA** PUBLICACIONES
ARBITRADAS
HUMANIDADES, SOCIALES Y CIENCIAS

CC+
COLECCIÓN
CIENCIA e
INVESTIGACIÓN

Sistemas inteligentes aplicados a la educación

RAMÓN ZATARAIN CABADA
MARÍA LUCÍA BARRÓN ESTRADA
KARINA MARIELA FIGUEROA MORA



**COMUNICACIÓN
CIENTÍFICA**

Sistemas inteligentes aplicados a la educación / Ramón Zatarain Cabada, María Lucía Barrón Estrada, Karina Mariela Figueroa Mora. — Ciudad de México : Comunicación Científica, 2023.

165 páginas : ilustraciones. — (Colección Ciencia e Investigación).

ISBN 978-607-59500-5-1

DOI 10.52501/cc.096

1. Sistemas tutoriales inteligentes. I. Zatarain Cabada, Ramón, autor. II. Barrón Estrada, María Lucía, autora. III. Figueroa Mora, Karina Mariela, autora. IV. Título. V. Serie.

LC: LB1028.73

Dewey: 371.334

D.R. Ramón Zatarain Cabada, María Lucía Barrón Estrada, Karina Mariela Figueroa Mora, 2023.

Primera edición en Ediciones Comunicación Científica, 2023

Diseño de portada e interiores: Arcelia Judith Bustillos Martínez

Ediciones Comunicación Científica S.A. de C.V., 2023

Av. Insurgentes Sur 1602, piso 4, suite 400,

Crédito Constructor, Benito Juárez, 03940, Ciudad de México,

Tel. (52) 55 5696-6541 • móvil: (52) 55 4516 2170

info@comunicacion-cientifica.com • www.comunicacion-cientifica.com

 comunicacioncientificapublicaciones  @ComunidadCient2

ISBN 978-607-59500-5-1

DOI 10.52501/cc.096



Esta obra fue dictaminada mediante el sistema de pares ciegos externos.
El proceso transparentado puede consultarse, así como el libro en acceso abierto,
en <https://doi.org/10.52501/cc.096>

Índice

<i>Resumen</i>	11
<i>Prefacio</i>	13

PRIMERA PARTE MARCO TEÓRICO Y CONCEPTUAL

1. Introducción a los sistemas inteligentes	19
1.1 Computación suave	20
1.2 Sistemas difusos	20
1.3 Redes neuronales	24
1.4 Algoritmos genéticos	30
1.5 Sistemas inteligentes híbridos	34
2. Computación educativa y tecnologías de aprendizaje	37
2.1 Aprendizaje electrónico	38
2.2 Aprendizaje basado en la web	40
2.3 Aprendizaje móvil	41
2.4 Aprendizaje híbrido	43
2.5 Sistemas administradores de aprendizaje	44
2.6 Sistemas tutores inteligentes	46
2.7 Otros sistemas inteligentes de aprendizaje	49
3. Modelos de estilos de aprendizaje	51
3.1 Modelo de estilos de aprendizaje de Gregorc	52
3.2 Teoría de inteligencias múltiples de Howard Gardner	53
3.3 Indicador de tipo de Myers-Briggs	55
3.4 Modelo de estilos de aprendizaje de Felder-Silverman	56

4. Computación afectiva	61
4.1 Definición de emoción y afecto	62
4.2 Categorización de las emociones	63
4.3 Sistemas reconocedores de emociones en rostros	64
4.4 Sistemas reconocedores de emociones en voz	68
4.5 Sistemas reconocedores multimodales	70

SEGUNDA PARTE
APLICACIONES Y CASOS PRÁCTICOS

5. EDUCA: una herramienta de autor para cursos de aprendizaje electrónico y móvil	75
5.1 ¿Qué es EDUCA?	76
5.2 Arquitectura de EDUCA	76
5.3 Creación de cursos en PC y visualización en dispositivos móviles	79
5.4 Motor de identificación de estilos de aprendizaje	83
5.5 Diseño de la red neuronal para clasificación de estilos de aprendizaje	87
5.6 Entrenamiento de la red neuronal	88
6. Zamná: una red social de aprendizaje adaptativo	93
6.1 La web 2.0	94
6.2 ¿Qué es Zamná?	95
6.3 Análisis y diseño de Zamná	97
6.4 Arquitectura de la red social educativa	99
6.5 Navegación en el sistema Zamná	102
7. Fermat: sistema para la enseñanza de matemáticas	111
7.1 La enseñanza de matemáticas en México	112
7.2 ¿Qué es Fermat?	115
7.3 Análisis, diseño e implementación de Fermat	116

7.4 Sistema experto difuso	127
7.5 Reconocimiento de emociones por medio de redes neuronales	130
8. Memorato: juego clásico en una época moderna	139
8.1 ¿Qué es Memorato?	140
8.2 Arquitectura de Memorato	141
8.3 Diagrama de flujo de Memorato	142
8.4 Versión multijugador	147
8.5 Sistema de recomendación	148
8.6 Experimentación	149
<i>Referencias</i>	151
<i>Sobre los autores</i>	159
<i>Índice de figuras</i>	161
<i>Índice de tablas</i>	164

Resumen

Las computadoras se han utilizado en la educación desde el siglo pasado con el fin de proporcionar a los estudiantes herramientas que contribuyan a su proceso de aprendizaje. El desarrollo de sistemas computacionales aplicados a la educación involucra diferentes áreas de conocimiento para poder generar recursos educativos adaptables a las necesidades personales de los estudiantes.

Esta obra se divide en dos secciones: *Marco teórico y conceptual* y *Aplicaciones y casos prácticos*, donde se describe el desarrollo de sistemas inteligentes aplicados a la educación.

La primera parte contiene cuatro capítulos dedicados a explicar los fundamentos básicos de algunas áreas de conocimiento que se requieren para desarrollar sistemas computacionales para distintos modelos de aprendizaje, los cuales funcionan en diversas plataformas y poseen características adaptativas de acuerdo con estilo de aprendizaje del estudiante o su estado emocional. La adaptación del contenido educativo, así como el reconocimiento de estados emocionales, se logran a través de la incorporación de técnicas de inteligencia artificial como la lógica difusa y las redes neuronales artificiales, entre otras.

La segunda parte describe cuatro trabajos de investigación que se desarrollaron aplicando diferentes modelos y técnicas para generar sistemas computacionales aplicados a la educación. Estos casos prácticos describen un sistema de autoreo, que es una herramienta computacional para que un profesor o instructor cree sus propios cursos o tutoriales inteligentes sin

necesidad de poseer amplios conocimientos en computación; una red social de aprendizaje donde la comunidad estudiantil puede interactuar utilizando los recursos educativos generados en la herramienta de autoreo; una aplicación web que incluye un sistema tutor inteligente y afectivo que adapta el contenido para cada estudiante, y finalmente un juego para dispositivos móviles que permite generar parejas de pregunta-respuesta para incorporarlas en un juego de memoria digital utilizando un módulo de recomendación con base en las preferencias del estudiante.

Los cuatro sistemas inteligentes que se presentan en la segunda parte tienen incorporados módulos inteligentes que hacen uso de lógica difusa, algoritmos genéticos o redes neuronales con el fin de proporcionar al usuario una experiencia de aprendizaje personalizada más adecuada a sus necesidades cognitivas y afectivas.

Palabras clave: *sistemas tutores inteligentes, aprendizaje electrónico, inteligencia artificial, emociones en el aprendizaje, computación afectiva.*

Prefacio

Los llamados “sistemas inteligentes” imitan algunas de las características de la inteligencia natural y ofrecen un enfoque metodológico definido para resolver problemas críticos y algo complicados y obtener soluciones consistentes y confiables en el tiempo. La flexibilidad, la adaptabilidad, la memoria, el aprendizaje, la dinámica temporal, el razonamiento y la capacidad de manejar información ambigua e imprecisa son características de la inteligencia de estos sistemas.

Los sistemas inteligentes se construyen para dar a las computadoras la capacidad de ver, oír y oler, analizar, aprender, tomar decisiones y estimular la creatividad humana. Algunos de sus campos de investigación son el aprendizaje máquina, el reconocimiento de patrones, la computación cognitiva, la computación inspirada en el cerebro y el análisis de *big data*. Hoy en día sus aplicaciones son numerosas, por ejemplo, en la atención médica y sanitaria (dermatología, oftalmología, psiquiatría, oncología, reumatología, endocrinología), en las ciencias naturales (astronomía, bioinformática), en seguridad (vigilancia por medio de visión artificial), en robótica y en educación. Es en este último importante campo de aplicación hacia donde se orienta nuestra principal línea de investigación y las aplicaciones que se presentan en este libro.

La estructura del libro se divide en dos grandes partes.

La primera parte, llamada *Marco teórico y conceptual*, presenta en cuatro capítulos la teoría y/o fundamentos básicos de las áreas del conocimien-

to que se requieren para la implementación de un sistema inteligente aplicado a la educación. En primer lugar, las técnicas o métodos inteligentes, como son los sistemas difusos, las redes neuronales, los algoritmos genéticos o la combinación de dos o más de ellas, conocidos como “sistemas híbridos”. En segundo lugar, las diferentes formas de aprendizaje usando la computadora, a saber el aprendizaje electrónico (*eLearning*), el aprendizaje basado en la web (*Web-based learning*), el aprendizaje móvil (*mLearning*), el aprendizaje híbrido, los sistemas administradores del aprendizaje (LMS por sus siglas en inglés), los sistemas de tutores inteligentes y otros sistemas o ambientes inteligentes de aprendizaje. En tercer lugar, los principales modelos o teorías de estilos de aprendizaje, como el de Gregorc, el de Gardner (inteligencias múltiples), el de Myer-Briggs y el de Felder-Silverman. Y, en cuarto lugar, los conceptos más importantes de la computación afectiva, es decir las definiciones y diferencias entre emoción y afecto, los sistemas inteligentes que detectan o reconocen emociones en el rostro, en la voz o en una combinación entre varios modos (multimodal).

La segunda parte, llamada *Aplicaciones y casos prácticos*, presenta en cuatro capítulos diferentes ejemplos de sistemas inteligentes aplicados en el ámbito de la educación. En primer lugar, se presenta **Educa**, una herramienta de autor para que un profesor o instructor cree sus propios cursos o tutoriales inteligentes sin requerir conocimientos de programación. Educa le permite al autor diseñar un curso que se adapte al mejor estilo de aprendizaje del estudiante, para lo cual se genera una red neuronal artificial que adapta los contenidos de aprendizaje. La red neuronal hace uso de técnicas de lógica difusa y de algoritmos genéticos para su mejor diseño e implementación. En segundo lugar, se presenta **Zamná**, una red social que permite a un grupo de estudiantes, padres y profesores administrar cursos o tutores inteligentes como los creados por Educa. En tercer lugar, se presenta **Fermat**, una aplicación web que se compone de dos componentes: una red social de aprendizaje y un sistema de tutor inteligente. El sistema tutor presenta contenidos de aprendizaje adaptándolos al estado emocional o afectivo del estudiante, el cual se reconoce usando una red neuronal que, por medio de extracción de características del rostro, reconoce la emoción actual del estudiante. Además, el tutor interactúa con el estudiante por medio de un agente pedagógico (un personaje virtual) que se comunica cons-

tantemente con él. Finalmente, **Memorato** es una aplicación multijugador para dispositivos móviles pensada para coadyuvar al conocimiento. El sistema permite personalizar el clásico juego de memoria. Un usuario debe crear pares sobre pregunta-respuesta, y posteriormente buscará ese par. Memorato ofrece un módulo de recomendación en la búsqueda de nuevos juegos basado en el historial del usuario y el área en la que jugó la última vez.

Los cuatro sistemas inteligentes que se presentan en la segunda parte tienen incorporados módulos inteligentes que hacen uso de lógica difusa, algoritmos genéticos o redes neuronales, con el fin de proporcionar al usuario una experiencia de aprendizaje personalizada más adecuada a sus necesidades cognitivas y afectivas.

Primera Parte

MARCO TEÓRICO Y CONCEPTUAL

1. Introducción a los sistemas inteligentes

A través de los años, diversos autores han propuesto diferentes definiciones para el concepto de “inteligencia artificial” (IA). Boden (1977) manifiesta que Minski la definió como “la ciencia de hacer que las máquinas realicen cosas que requerirían inteligencia si fueran hechas por humanos” (p. 394). Más recientemente, Copeland (2022) define el concepto en la *Enciclopedia británica* como “la habilidad de una computadora digital o de un robot controlado por computadora para realizar tareas comúnmente asociadas a seres inteligentes”. Con tal diversidad de definiciones, se puede concluir que la inteligencia artificial es el conjunto de métodos y herramientas computacionales que se utilizan para resolver problemas que requieren inteligencia humana.

Los sistemas inteligentes son sistemas de *software* que manifiestan un comportamiento inteligente con un enfoque metodológico para resolver problemas complejos de IA en donde se obtienen buenos resultados. Un sistema inteligente emula aspectos exhibidos por el ser humano ante la naturaleza, como son, entre otros, aprendizaje, razonamiento, capacidad de comunicación, tolerancia y adaptabilidad a la imprecisión e incertidumbre.

Un sistema inteligente puede ser implementado con un enfoque o método de computación dura (menos común) o de computación suave (*soft computing*). En el enfoque “duro”, para resolver problemas predomina el uso de lógica binaria (*crisp*), análisis numérico, modelos probabilísticos, pro-

gramación matemática y funcional, etc. El enfoque “suave” se enfoca más en el análisis y diseño de sistemas inteligentes usando principalmente técnicas de redes neuronales, lógica difusa, algoritmos genéticos y razonamiento probabilístico.

1.1 Computación suave

Algunos problemas representan tal grado de complejidad que requieren el uso de diferentes técnicas inteligentes como lógica difusa, redes neuronales y algoritmos genéticos. Según Negnevitsky (2005), cuando se combinan varias de estas tecnologías para implementar sistemas inteligentes híbridos que sean capaces de razonar y aprender en ambientes inciertos e imprecisos, se hace referencia al concepto de “computación suave”.

A diferencia de la inteligencia artificial convencional, que trabaja y manipula números (datos duros) y símbolos principalmente, la computación suave opera con palabras (datos suaves) que permiten trabajar de una manera más natural con información incierta, imprecisa e incompleta, lo cual es muy común en problemas de la vida real. En otras palabras, la computación suave intenta modelar nuestra forma de razonar, aprender, comunicarnos y tomar decisiones por medio de un grupo de tecnologías inteligentes. A continuación, se presentan tres de las más populares: sistemas difusos, redes neuronales y algoritmos genéticos.

1.2 Sistemas difusos

Historia

Los sistemas difusos se basan en la lógica difusa (*Fuzzy Logic*), también conocida como “lógica borrosa”, la cual fue inicialmente propuesta en los años 30 por Jan Lukasiewicz (1930), con el nombre de “lógica de posibilidades”. En su trabajo, Lukasiewicz introdujo una lógica que extendía los valores de la verdad a un rango similar al de los números reales entre cero y uno. Para esto, un valor en este rango representaba la posibilidad de que

un evento fuera verdadero o falso. Por ejemplo, una temperatura de 40 grados Celsius podía ser considerada “caliente” con 0.92 de valor de posibilidad.

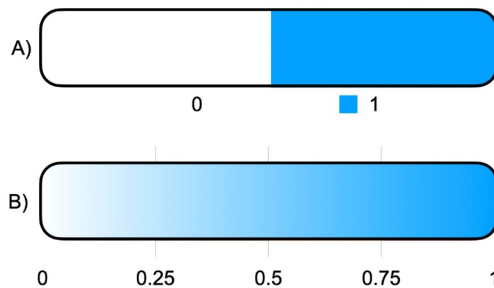
Poco después, en 1937 Max Black publicó el artículo titulado “Vaguedad: Un ejercicio de análisis lógico”. En este trabajo, Black argumentaba que un valor continuo (*continuum*) implicaba un grado. Él con su trabajo expresaba que la vaguedad es un concepto también sobre probabilidades, lo cual explicaba con el ejemplo de que si tenemos un tronco de madera el cual poco a poco se va convirtiendo hasta llegar a ser un sillón de lujo, podríamos referirnos a cada transformación del tronco como un grado que va incrementándose hasta convertirse en el sillón. En el apéndice, Black proponía una lógica de conjuntos difusos, tal vez su mayor contribución.

Finalmente, en 1965 estos trabajos fueron extendidos por Lofti Zadeh para crear así un sistema formal de lógica matemática, el cual fue publicado en su famoso artículo “Conjuntos difusos (Fuzzy Sets)”.

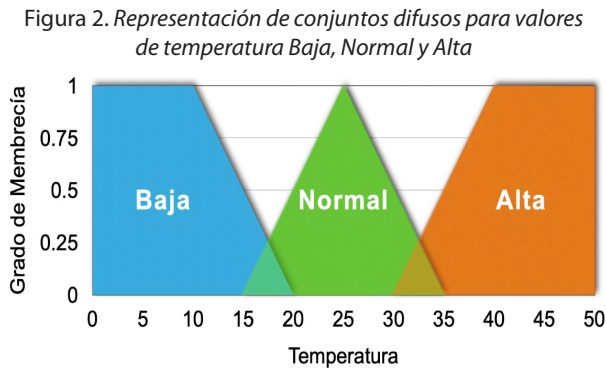
Fundamentos

La lógica difusa (*Fuzzy Logic*) viene a ser un conjunto de principios matemáticos para representar el conocimiento basado en grados de membresía. A diferencia de una lógica tradicional —como la lógica booleana, que solo admite dos únicos valores (cero y uno)—, la lógica difusa opera con un rango entre 2 valores, que normalmente son el cero y el uno. En la figura 1 se puede ver de forma gráfica esta diferencia.

Figura 1. Rango de los valores en a) un sistema booleano (0 y 1) y b) un sistema difuso con un número infinito de valores



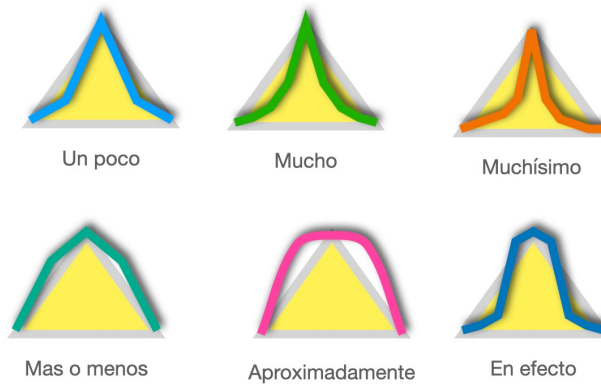
Un conjunto difuso es un conjunto con límites vagos, donde los conjuntos pueden tener nombres, por ejemplo, para referirse a la temperatura de una temporada es posible representar conjuntos llamados **Baja**, **Normal** o **Alta**, como se muestra en la figura 2. Un conjunto difuso en una computadora se representa como una función que transforma los elementos de un conjunto (por ejemplo, temperatura **Baja**), en un valor respectivo en grados de membresía (eje vertical). Típicamente las funciones de membresía usadas en los sistemas expertos difusos son representadas por triángulos y trapecios (véase figura 2).



La lógica difusa se compone también de variables lingüísticas como **Temperatura** y valores lingüísticos como **Baja**, los cuales se usan para describir términos o conceptos con valores vagos o difusos. Esos valores se representan después en los conjuntos difusos.

Los cercos (*hedges*) son calificadores de conjuntos difusos usados para modificar la forma del conjunto difuso, esto con el fin de dar una mayor flexibilidad y/o amplitud de expresión del lenguaje humano. Los cercos incluyen adverbios tales como: muy, más o menos, algo, más, menos, etc. Con estos cercos se llevan a cabo operaciones matemáticas de concentración por medio de la reducción del grado de membresía de elementos difusos. Por ejemplo, el cerco de la figura 3 ubicado arriba a la izquierda se puede referir a una temperatura “un poco normal” o a una presión “un poco alta”, mientras que el cerco ubicado abajo a la izquierda corresponde a una estatura “más o menos alta”, etc.

Figura 3. Diferentes cercos (hedges) para un conjunto difuso como temperatura normal



Las reglas difusas son usadas para capturar el conocimiento humano y para hacer inferencias o razonamientos. Una regla difusa es un estatuto condicional de la forma:

$$\begin{array}{ll} \text{IF} & X \text{ is } A \\ \text{THEN} & Y \text{ is } B \end{array}$$

donde X y Y son variables lingüísticas y A y B son valores lingüísticos determinados por los conjuntos difusos.

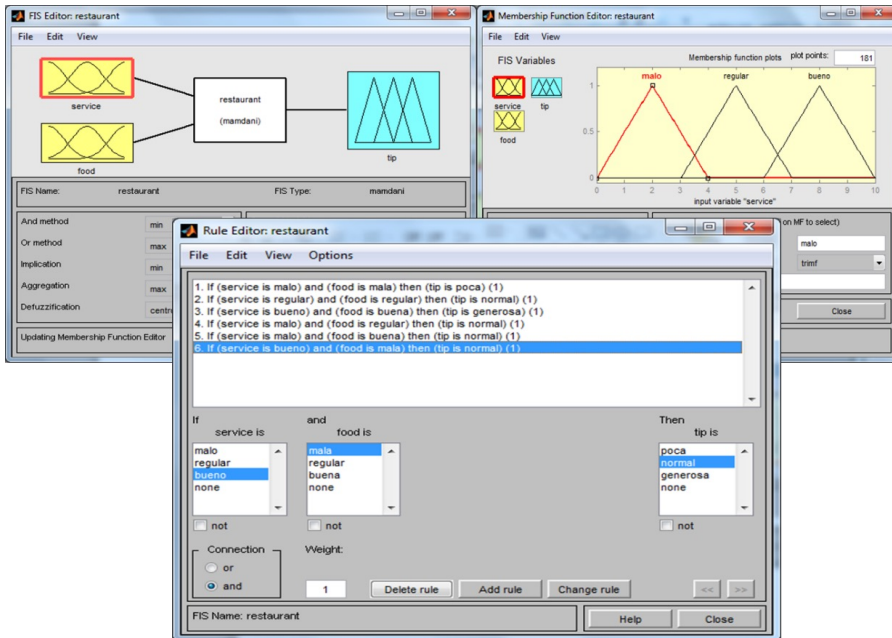
Por ejemplo:

$$\begin{array}{ll} \text{IF} & \text{TEMPERATURA} = \text{ALTA} \\ \text{THEN} & \text{REFRIGERACION} = \text{ENCENDIDO} \end{array}$$

Ejemplo

Un problema usado comúnmente como ejemplo de un sistema difuso es conocido como “*The Tipping Problem*” y consiste en determinar el monto apropiado de propina en un restaurante. A continuación, se describe un pequeño sistema inteligente difuso que apoya a un cliente en tomar la decisión del monto de propina que deberá pagar en un restaurante considerando dos aspectos: la calidad de la comida y la calidad del servicio recibido. El sistema fue implementado usando la herramienta Matlab 7. En la figura 4 se muestran las interfaces en Matlab para el sistema difuso completo.

Figura 4. Sistema difuso para obtener la propina en un restaurante



En la parte de arriba a la izquierda, aparecen las dos variables difusas o lingüísticas de entrada **service** (servicio) y **food** (comida), y la variable difusa de salida **tip** (propina). Por otro lado, en la sección superior derecha se observa que se definieron 3 valores difusos: **malo**, **regular** y **bueno** para la variable **service**, y en la ventana del medio aparecen las reglas difusas que evalúan el servicio y la comida para obtener la propina (**poca**, **normal** o **generosa**). Como se puede apreciar en la figura, la primera regla define que, si el servicio y la comida son malos, la propina es poca, y por el contrario, la regla 3 establece que, si el servicio y la comida son buenos, la propina es generosa.

1.3 Redes neuronales

Historia

Las redes neuronales forman parte de un campo más amplio llamado “aprendizaje máquina”. El aprendizaje máquina se enfoca en el diseño y

desarrollo de algoritmos que permiten que las computadoras aprendan, usando para ello información proveniente de sensores o bases de datos.

Las primeras investigaciones generadas alrededor de las redes neuronales datan de 1943 (Haykin, 1999) con el neuropsicólogo Warren McCulloch y el matemático Walter Pitts con la teoría McCulloch-Pitts de *redes neuronales formales*. En ese tiempo se crearon grandes expectativas alrededor de esta nueva tecnología de aprendizaje máquina. Sin embargo, en el período 1969-1981 hubo una decepción general respecto a los reducidos éxitos de la inteligencia artificial, y en específico con las redes neuronales (Haykin, 1999). Fue hasta mediados de los años 80 cuando el interés en las redes neuronales volvió debido en gran medida al trabajo reportado por los investigadores Rumelhart, Hinton y Williams con el desarrollo de un algoritmo de propagación hacia atrás (*back-propagation*).

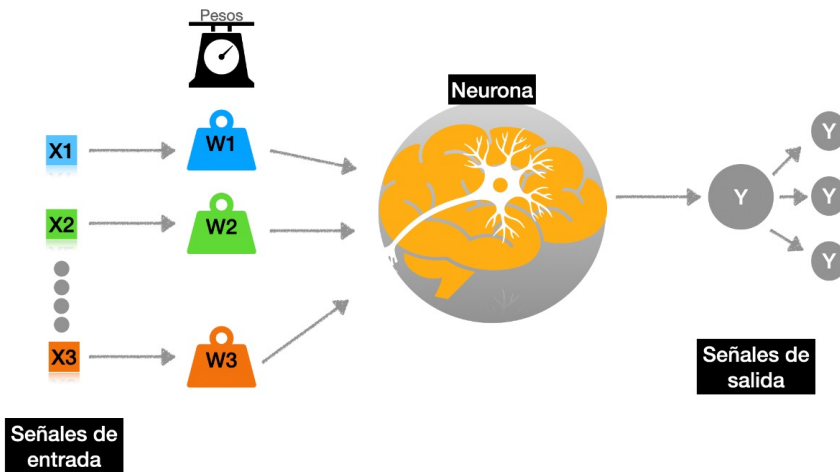
Fundamentos

El cerebro humano posee un aproximado de 10^{11} pequeñas unidades conocidas como neuronas. A su vez, las neuronas se encuentran conectadas densamente entre sí, formando un estimado de 10^{15} enlaces, llamados sinapsis (Munakata, 2008). Una red neuronal artificial es un modelo computacional que intenta replicar parcialmente la funcionalidad del cerebro humano. Las redes neuronales artificiales, de forma muy similar al cerebro, están compuestas de neuronas artificiales y de conexiones entre ellas. La idea principal es la de imitar los procesos de comunicación entre las neuronas y su propiedad de elasticidad, fortaleciendo y debilitando las conexiones entre las mismas.

Las neuronas en este tipo de redes suelen encontrarse altamente conectadas entre sí. Cada neurona tiene un conjunto de entradas y un conjunto de salidas, que a su vez se conectan, en forma de entradas, a otras neuronas, y en conjunto se pasan información a través de las conexiones existentes entre ellas. Cada neurona emite solo una información distinta a todas las neuronas que están conectadas a ella. Sin embargo, las conexiones entre las neuronas poseen distintos pesos que determinan la intensidad con la que se recibe la información en otra neurona. De esta forma, la misma información enviada desde una neurona puede ser percibida con una fuerza distin-

ta por diferentes neuronas, dependiendo del peso que media entre las conexiones de la emisora y las receptoras. En la figura 5 se muestra el diagrama de una neurona artificial con un conjunto de entradas, pesos y salidas.

Figura 5. Diagrama de una neurona



Las redes neuronales suelen organizarse en capas, las cuales se comunican únicamente con las capas adyacentes. De esta forma, las neuronas que se encuentran en la capa de entrada solo pueden comunicarse con las que están en la capa oculta. Por otra parte, las que se encuentran en la capa oculta pueden comunicarse con la capa de entrada y la capa de salida.

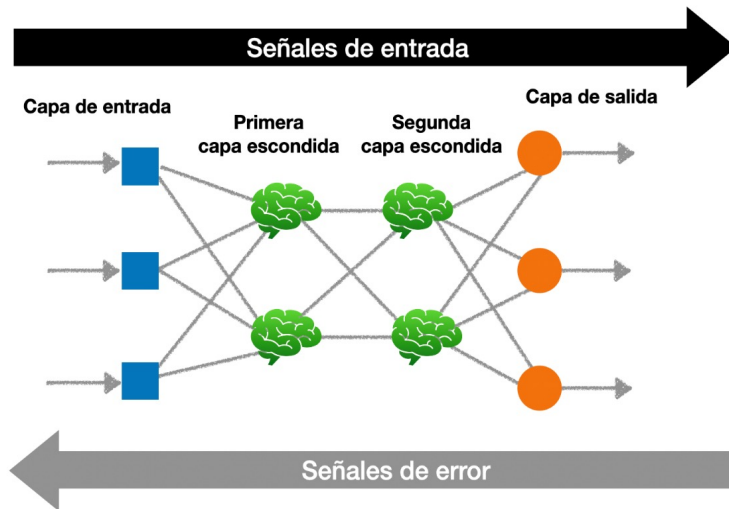
El modelo computacional que define a las redes neuronales está compuesto por los parámetros siguientes: el tipo de neuronas, la arquitectura de la red y el algoritmo de aprendizaje (Kasabov, 1998).

Los tipos de conexiones que presentan las neuronas entre sí se definen como la topología de la red. Una red neuronal puede estar totalmente conectada —es decir, cada neurona está conectada con el resto— o parcialmente conectada, cuando las neuronas están conectadas solo con otras neuronas en diferentes capas. El patrón de conexiones entre las neuronas se conoce generalmente como la arquitectura de la red neuronal. De acuerdo con el número de neuronas de entrada y salida y con el número de capas de la red, es posible reconocer dos arquitecturas de conexión distintas: la autoasociativa, en la que las neuronas de entrada también son neuronas de

salida, y la heteroasociativa, en la que existen dos conjuntos de neuronas: uno de entrada y otro de salida.

Otra clasificación de las redes neuronales de acuerdo con su arquitectura se obtiene considerando las conexiones de retroalimentación de la red. Si la red neuronal presenta conexiones de retroalimentación, tiene una arquitectura *feedback*. Las conexiones van de las neuronas de salida a las de entrada, permitiendo que la red tenga un registro de sus estados previos. Por lo tanto, el siguiente estado de la red no depende únicamente de la señal de entrada, sino también de los estados previos de la red. Cuando la red neuronal no posee conexiones de retroalimentación, entonces tiene una arquitectura *feedforward* o hacia adelante y propagación de errores hacia atrás (figura 6). Las redes neuronales con esta arquitectura no son capaces de recordar sus estados previos y sus salidas dependen solamente de sus entradas.

Figura 6. Red neuronal artificial feedforward



La habilidad de las redes neuronales para aprender se logra gracias a que utilizan algoritmos de entrenamiento, los cuales son llamados también “algoritmos de aprendizaje”. Los algoritmos de entrenamiento de las redes neuronales se pueden clasificar en tres categorías: aprendizaje supervisado, aprendizaje no-supervisado y aprendizaje por reforzamiento. El uso de cada uno de estos algoritmos suele ser independiente de la arquitectura de la red neuronal.

Aprendizaje supervisado. Cuando se utilizan algoritmos de aprendizaje supervisado, la red neuronal artificial recibe un conjunto de datos de entrenamiento conocido como corpus o *dataset*, en donde los ejemplos de entrenamiento están compuestos por un conjunto de vectores de entrada \mathbf{x} y un conjunto de vectores de salida \mathbf{y} (ver ejemplo en Figura 7). El proceso de entrenamiento se realiza hasta que la red neuronal aprenda a asociar el vector \mathbf{x} con la salida correspondiente \mathbf{y} .

Tareas como el reconocimiento de patrones y regresiones se realizan utilizando aprendizaje supervisado. También, aquellas tareas que tienen relación con información secuencial, como el reconocimiento del habla y de gestos. Se suele hacer la analogía de estar aprendiendo con un profesor, ya que se provee retroalimentación continua a las soluciones que se van obteniendo.

Aprendizaje no-supervisado. Cuando se usan algoritmos de aprendizaje no-supervisado solo se requiere el vector de entrada \mathbf{x} , porque la red neuronal aprende las características de los vectores que se le presentan. A la red neuronal se le proporciona una función de costo con el objetivo de minimizarla. Esta función de costo es dependiente de la tarea que se está resolviendo. Con este tipo de algoritmos de aprendizaje generalmente se abordan problemas de estimación: agrupamiento, distribuciones estadísticas, compresión y filtrado.

Un tipo popular de aprendizaje sin supervisión es el aprendizaje competitivo. En el aprendizaje competitivo, las neuronas compiten entre ellas para ser activadas. La neurona de salida que gana la competencia es conocida como la neurona ganadora-toma-todo.

A pesar de que el aprendizaje competitivo fue propuesto a principios de los años 70, fue prácticamente ignorado hasta los 80, cuando Teuvo Kohonen presentó una clase especial de redes neuronales artificiales llamadas “mapas de características auto-organizadas” (*Self-Organizing Feature Maps*) (Kohonen, 1990). Los mapas autoorganizados (SOM por sus siglas en inglés) presentan un excelente rendimiento una vez que han sido entrenados. Además, los SOM, también conocidos como “mapas de Kohonen”, convergen rápidamente en su etapa entrenamiento y requieren de un conjunto de datos relativamente pequeño. Es importante mencionar que requieren poca cantidad de memoria y tienen buena velocidad en tiempo de ejecución. Las

redes SOM se utilizan para diversas aplicaciones, entre ellas el agrupamiento de datos de acuerdo con ciertas características.

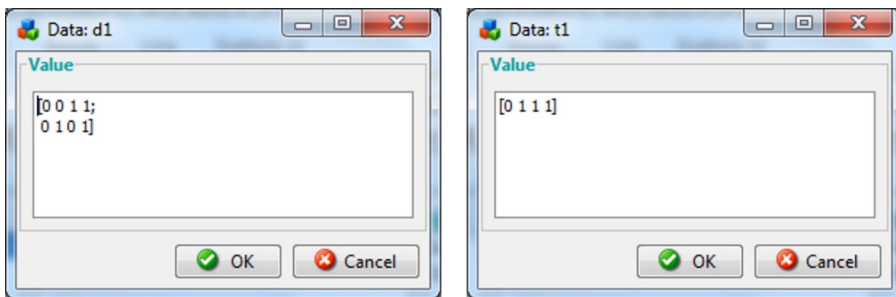
Aprendizaje por reforzamiento. El aprendizaje por reforzamiento es una combinación de los otros dos enfoques. A este tipo de algoritmos de aprendizaje también se le conoce como “aprendizaje de recompensa-penalización”. Algunas tareas relacionadas con este tipo de algoritmos son los problemas de control, juegos y tareas de toma de decisiones secuenciales.

Ejemplo

A continuación se presenta una red neuronal artificial del tipo *perceptrón* para la operación lógica OR, implementada usando la herramienta Matlab.

La figura 7 muestra los datos de entrada para que la red sea entrenada (parte izquierda) y los datos de la salida deseada (parte derecha). Los datos de entrada son dos vectores que contienen las combinaciones posibles de los dos valores de entrada para la compuerta OR, mientras que para la salida se proporciona el vector que contiene el resultado de la compuerta OR para cada pareja de datos de entrada.

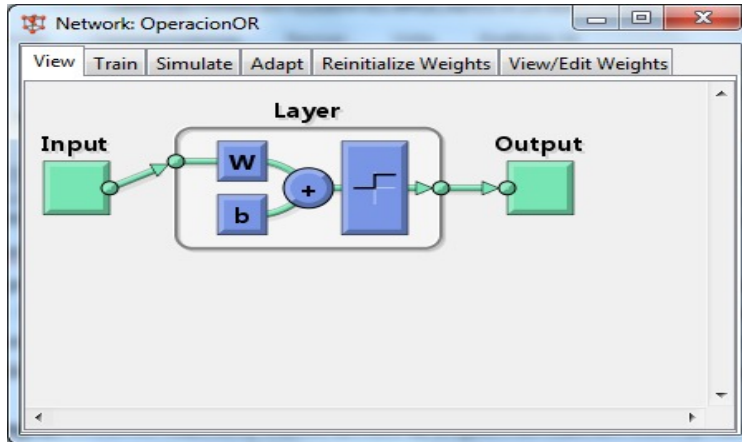
Figura 7. Datos de entrada y salida deseada de la red neuronal



En la figura 8, por su parte, se presenta la red neuronal *feedforward* con tres capas; donde la capa de entrada tiene dos neuronas de alimentación, y tanto la capa del medio (llamada también “capa escondida”) como la capa de salida tienen solo una neurona cada una. La neurona va a realizar un cálculo usando dos valores de entrada (cualquier combinación de los valo-

res cero y uno) y generará una salida de un solo valor (resultado de la operación OR) que será un uno o cero.

Figura 8. Red neuronal artificial perceptrón para operación lógica OR



La red neuronal inicia un proceso iterativo de entrenamiento (operación realizada en la pestaña “Train”) donde mediante ajuste a los valores de los pesos (*weights*) iniciales y cálculo de la salida actual en la neurona artificial, el algoritmo de entrenamiento busca que la salida actual de la red neuronal sea igual o tenga un valor muy cercano al de la salida deseada. Cuando esto sucede, el entrenamiento finaliza y se dice que la red “ya terminó de aprender”. Después de esto la red neuronal ya está preparada para realizar operaciones lógicas OR (se usa la pestaña “Simulate”).

1.4 Algoritmos genéticos

Historia

El origen de los algoritmos genéticos se remonta a principios de la década de 1950. En esta época, un grupo de científicos de las áreas de computación y biología colaboraron para desarrollar una simulación del comportamiento de una clase de procesos biológicos. Sin embargo, fue hasta los inicios de los años 70 que John Holland y sus colegas introdujeron el concepto de

“algoritmos genéticos” (Holland, 1975). Su objetivo era hacer que las computadoras hicieran lo que hace la genética. Un algoritmo genético, según la concepción de Holland, va comportándose en una forma progresiva tomando como base una población que va evolucionando por medio de “selección natural” y técnicas conocidas en la genética con el nombre de cruzamiento (*crossover*) y mutación.

Fundamentos

Los algoritmos genéticos (AG) son una clase de algoritmos de búsqueda no-determinística basada en evolución biológica (Negnevitsky, 2005). Dicho de manera más formal, un algoritmo genético es un proceso iterativo, el cual inicia con una población N y va generando continuamente nuevas y mejores poblaciones, usando para ello una función de evaluación de los cromosomas. Los miembros de la población son cromosomas individuales de acuerdo con el problema establecido. La función determina cuáles cromosomas son buenos y cuáles no. El área principal de aplicaciones de los AG es la búsqueda de soluciones óptimas en grandes espacios de búsqueda.

Un AG básico aplica los siguientes pasos:

- I. Representar el dominio de la variable del problema como un cromosoma de longitud fija, escoger el tamaño de una población de cromosomas N , la probabilidad de cruzamiento P_c y la probabilidad de mutación P_m . Un cromosoma artificial se define como una cadena binaria que representa los “genes”.
- II. Definir una función de aptitud (*fitness*) para medir el desempeño, adaptación o adecuación, de un cromosoma individual en el dominio del problema. La función de aptitud establece la base para seleccionar cromosomas que serán emparejados durante la reproducción.
- III. Generar aleatoriamente una población inicial de cromosomas de tamaño N : x_1, x_2, \dots, x_n .
- IV. Calcular la aptitud de cada cromosoma: $F(x_1), F(x_2), \dots, F(x_n)$.
- V. Seleccionar un par de cromosomas de la población actual para aparearse; los cromosomas padres son seleccionados con una probabilidad relacionada con su adaptación. Los cromosomas con gran

adaptabilidad tienen una mayor probabilidad de ser seleccionados para aparearse que los de baja probabilidad.

- VI. Crear un par de cromosomas descendientes (*offspring*) para aplicar los operadores genéticos (cruzamiento y mutación).
- VII. Colocar los cromosomas descendientes creados como la población nueva.
- VIII. Repetir el paso V hasta que el tamaño de la nueva población de cromosomas sea igual al tamaño de la población inicial N .
- IX. Reemplazar la población de cromosomas inicial (padres) con la nueva población (descendientes).
- X. Ir al paso IV y repetir el proceso hasta que el criterio de terminación sea satisfecho.

Un AG representa un proceso iterativo y cada iteración se conoce como generación. Un número típico de generaciones para un AG simple va de 50 a 500. El conjunto completo de generaciones es llamado corrida (*run*). Al final de la corrida se espera encontrar uno o más cromosomas altamente adaptables.

Ejemplos

Un ejemplo sencillo de aplicación de algoritmos genéticos, presentado por Negnevitsky (2005), es buscar o encontrar el valor máximo de la función $15x - x^2$, donde x toma valores enteros del 0 al 15.

Para este problema se define una población inicial de 6 cromosomas de tamaño 4 (genes), los cuales son generados aleatoriamente (note que esto representa un posible valor solución). A continuación, aparecen los 6 cromosomas de la población inicial:

X1: 1100

X2: 0100

X3: 0001

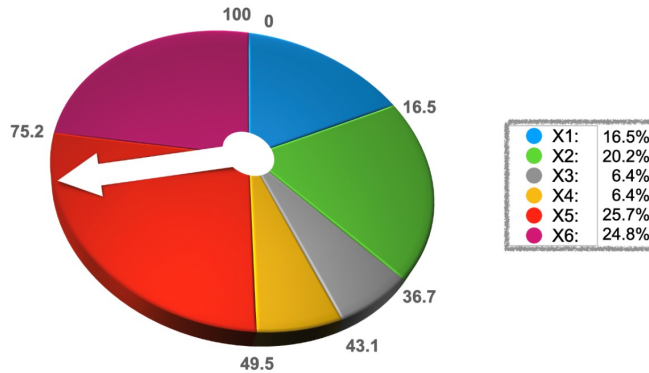
X4: 1110

X5: 0111

X6: 1001

Tomando la función $15x - x^2$ como función de aptitud y evaluando uno a uno los 6 respectivos cromosomas (Xs), se genera un porcentaje o radio para cada cromosoma (véase tabla en figura 9). Para seleccionar pares de cromosomas para cruzarse, se puede utilizar una técnica llamada de “selección por ruleta” (*roulette wheel selection*) que se muestra en la figura 9.

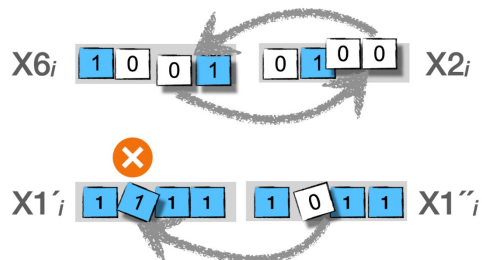
Figura 9. Selección por técnica de ruleta



Una vez que se seleccionan tres pares de cromosomas por medio de la ruleta y que se realizan las operaciones de cruzamiento y mutación, los cromosomas que se obtienen pasan a formar la nueva población.

En la parte superior de la figura 10 se muestra como dos cromosomas cruzan sus respectivos genes (01 con 00), mientras que en la parte inferior se muestra una mutación de un gene (1 por 0) en un cromosoma. Estas dos operaciones, cruzamiento y mutación, generan una nueva población, la cual nuevamente es evaluada por medio de la función de aptitud para entonces seleccionar nuevamente 3 pares de cromosomas. Al final de cierto número de generaciones quedará una población con solo cromosomas 0111 y 1000, que representan los valores máximos para la función $15x - x^2$.

Figura 10. Operaciones de cruzamiento (arriba) y mutación (abajo)



1.5 Sistemas inteligentes híbridos

Los tres sistemas inteligentes estudiados (sistemas difusos, redes neuronales artificiales y algoritmos genéticos) cuentan con fortalezas, pero también con debilidades, y cada uno de ellos tiene su propia área de aplicación.

Los sistemas difusos son una excelente opción para problemas en donde se requiere representar conocimiento, manipular información imprecisa o incierta, además de comunicarse con los usuarios en forma muy expresiva y natural. Sin embargo, son ineficientes respecto a problemas donde se requiera aprender y adaptarse a procesos nuevos. Por otra parte, las redes neuronales artificiales son la mejor opción para trabajar con problemas que requieren aprendizaje y adaptabilidad, pero no son una buena opción para representar conocimiento y son una especie de caja negra, por lo que difícilmente manejan bien aspectos de explicación o comunicación con el usuario. Por último, los algoritmos genéticos son la mejor opción para problemas de optimización y representan bien el conocimiento de un experto, pero no son buenos en descubrir nuevo conocimiento.

Desafortunadamente, en el mundo real los problemas que pueden ser resueltos mediante técnicas de sistemas inteligentes son de tal complejidad que difícilmente una sola tecnología es suficiente, y por ello se requiere la combinación de las fortalezas de dos o más tecnologías inteligentes. Este tipo de sistemas se conocen como “sistemas inteligentes híbridos”.

Ejemplos

Los sistemas neuroexpertos combinan una base neuronal de conocimientos, un extractor de reglas y una máquina de inferencias, todo esto dentro de un sistema experto. Una ventaja de estos sistemas es que para hacer el trabajo que hace un experto no requieren revisar todas las reglas necesarias, que es lo que hace un sistema experto tradicional. La base neuronal de conocimientos ayuda a extraer solo las reglas necesarias para tomar decisiones.

Otro ejemplo son los sistemas neurodifusos, que mezclan un conjunto de reglas difusas junto con su manejo de inferencias dentro de una red neuronal de cinco capas. Las ventajas que contienen las redes neuronales

en cuanto a aprendizaje y adaptabilidad se suman a las ventajas de representación y comunicación de conocimiento e información de un sistema difuso.

Un tercer ejemplo de sistema inteligente híbrido son las redes neuroevolutivas, las cuales aplican un algoritmo genético para mejorar iterativamente una red neuronal. Esto es muy importante debido a que el diseño de una red neuronal (topología, número de capas, pesos entre neuronas, etc.) es un trabajo más artesanal que de ingeniería. La fortaleza de un algoritmo genético para encontrar soluciones óptimas puede ayudar a encontrar la mejor red neuronal artificial para un problema dado.

2. Computación educativa y tecnologías de aprendizaje

Las computadoras se han utilizado en la educación desde finales de la década de 1960, con PLATO (*Programmed Logic for Automated Teaching Operations*) en la Universidad de Illinois (Molnar, 1990; Silvern, 1983), que se utilizó en escuelas primarias, de pregrado y por estudiantes de colegios comunitarios. Sin embargo, fue el lenguaje de programación Logo el que inició un nuevo campo del aprendizaje llamado “computación educativa”. Logo se desarrolló a principios de la década de 1970 para alentar a los estudiantes a pensar rigurosamente en matemáticas mediante el uso de ellas para construir productos significativos, como dibujos y procesos (Papert, 1980). Debido a que Logo era fácil de usar, los estudiantes podían expresar fácilmente los procedimientos para tareas simples.

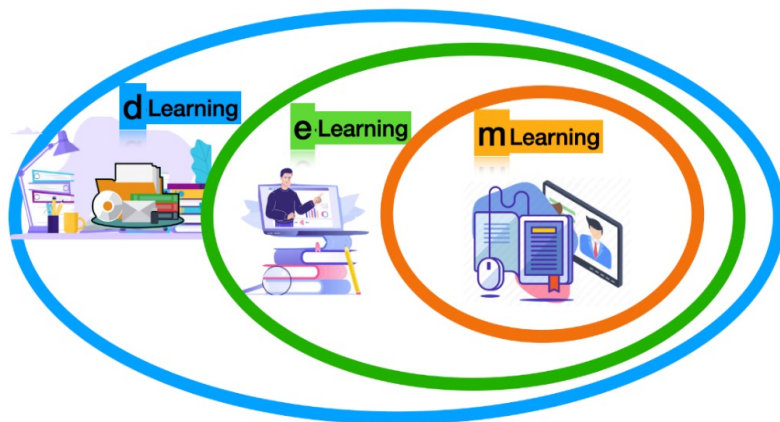
La educación basada en computadoras puede mejorar el aprendizaje en distintos niveles de educación, desde etapas tempranas como preescolar hasta nivel superior o posgrado. Desde hace más de 30 años, estudios científicos bien controlados han demostrado que los estudiantes que usaron sistemas de aprendizaje basados en computadoras aumentaron sus puntajes entre un 10 y un 20%, disminuyeron el tiempo para alcanzar las metas en un tercio y mejoraron su rendimiento en clase en aproximadamente la mitad de la desviación estándar (Kulik & Kulik, 1991).

Por otra parte, en las últimas décadas el desarrollo de tecnologías como el internet y la web, los dispositivos móviles como las tabletas y los teléfonos celulares, las plataformas y los sistemas administradores del aprendizaje y,

de manera especial, los sistemas inteligentes, han revolucionado la forma en que los estudiantes adquieren nuevos conocimientos a cualquier hora, en cualquier lugar y de cualquier forma (*anyplace, anytime, anyway*). Además, el aprendizaje puede ser personalizado o inteligente.

En este capítulo se presentan diferentes modalidades o métodos de aprendizaje basados en el uso de la computadora y el *software*. En la figura 11 se encuentran representados tres tipos de aprendizaje no presencial: aprendizaje a distancia (*dLearning*), aprendizaje electrónico (*eLearning*) y aprendizaje móvil (*mLearning*). El aprendizaje a distancia se refiere a cualquier forma de aprendizaje donde no es requisito que el estudiante se encuentre presente físicamente en algún lugar y no requiere de dispositivos electrónicos. El aprendizaje electrónico es un subconjunto del aprendizaje a distancia, así como el aprendizaje móvil es un subconjunto del aprendizaje electrónico.

Figura 11. Tres tipos de aprendizaje no presencial



2.1 Aprendizaje electrónico

El término “aprendizaje electrónico” (*eLearning*) es muy amplio y suele utilizarse para describir cualquier tipo de ambiente de aprendizaje que tenga relación con el uso de computadoras. En algunos casos, el aprendizaje electrónico ha sido definido como pedagogía impulsada por tecnología digital (Nichols, 2008). Aunque también se refiere a los medios utilizados para

implementar el proceso de enseñanza, que pueden haber sido diseñados para diferentes modelos educativos (presencial, a distancia, etc.) y distintas filosofías de la práctica educativa (constructivismo, conductismo, etc.).

La Comisión Europea define al aprendizaje electrónico (*eLearning*) como “el uso de nuevas tecnologías multimedia e Internet para mejorar la calidad del aprendizaje facilitando el acceso a recursos y servicios, así como los intercambios y la colaboración a distancia” (European-Commission, 2001, p. 2).

Actualmente, en el ámbito educativo, el aprendizaje electrónico integra las tecnologías de la información y la comunicación (TIC) como una herramienta de aprendizaje que utiliza principalmente internet como canal de distribución de contenidos en donde la interacción no siempre es presencial ni en un horario fijo. Otro aspecto importante es que se utilizan especialmente plataformas que comparten contenidos y comunican a los usuarios (estudiantes y profesores).

El aprendizaje electrónico tiene como característica la gran variedad de presentaciones posibles; puede ser presencial, móvil, a distancia o una combinación de los anteriores. Ofrece flexibilidad en el lugar y el momento en el cual se lleva a cabo. Esta característica, o funcionalidad, está disponible tanto para el profesor como para el estudiante (Ally, 2004). Esto implica que una mayor cantidad de estudiantes puede tener acceso a material didáctico diseñado por profesores que trabajan en grupos colaborativos. Además, la cantidad y variedad de contenidos a los que es posible acceder a través del aprendizaje electrónico es difícilmente igualado por cualquier otro tipo de aprendizaje (Oblinger & Hawkins, 2005).

El aprendizaje electrónico tiene la ventaja de adaptarse a tópicos complejos y de interés especial para el estudiante. El éxito o fracaso del uso del *eLearning* recae en tres aspectos importantes: 1) el contenido y la forma en cómo se presenta al usuario, 2) el enfoque metodológico orientado al tipo de estudiantes a quien va dirigido y 3) el enfoque tecnológico, el cual se refiere a las características tecnológicas necesarias para la correcta implementación del proyecto (Rodríguez & Caro, 2003; Salinas, 2005).

Existen varias ventajas al usar el aprendizaje electrónico; por ejemplo, se pueden realizar prácticas con retroalimentación, pues el material que se presenta puede ser interactivo con el estudiante; se puede trabajar en forma

colaborativa dependiendo si el *software* permite que varios usuarios se puedan comunicar entre sí; la instrucción puede ser personalizada o adaptada a las necesidades del estudiante si el *software* cuenta con métodos inteligentes para realizar el aprendizaje inteligente, y también permite aprender de situaciones complejas y riesgosas al usar *software* de tipo simulación en ciertos casos.

2.2 Aprendizaje basado en la web

El aprendizaje basado en la web (WBL por sus siglas en inglés) engloba todas las intervenciones educativas que hacen uso de internet (o una intranet local) y puede ser considerado también como un tipo de aprendizaje electrónico, con la particularidad de que los contenidos de aprendizaje se encuentran en páginas web. En el aprendizaje electrónico eso no siempre sucede pues los contenidos pueden estar en un medio electrónico, como una computadora. El aprendizaje basado en web incluye: cursos en línea (síncronos y asíncronos), tutorías en línea, grupos de discusión en línea, simuladores virtuales, entre otros. Cada una de estas modalidades tiene sus propias características, así como ventajas y desventajas; además hoy en día existen herramientas y plataformas que facilitan el uso de cada una de estas y es posible utilizar una combinación de dos o más de ellas.

El WBL incluye contenido de cursos en línea, aunque el material de los cursos basados en la web se diseña e implementa por medio de páginas web ya sea estáticas o dinámicas. Una de las ventajas de usar la web para acceder a los contenidos de aprendizaje es que las páginas pueden tener hipervínculos a otras páginas de la web, lo que permite el acceso a una gran cantidad de información basada en la web (McKimm *et al.*, 2003).

El WBL puede tomar la forma de (1) un aprendizaje en línea puro en el que el plan de estudios y el aprendizaje se implementan en línea sin una reunión cara a cara entre el instructor y los estudiantes, o (2) un híbrido en el que el instructor se encuentra con los estudiantes la mitad del tiempo en línea y la mitad del tiempo en el aula, dependiendo de las necesidades y requisitos del plan de estudios. El aprendizaje basado en la web se puede integrar en un plan de estudios que se convierte en un curso com-

pleto en sí mismo o como complemento de cursos tradicionales. Además, el aprendizaje basado en la web también puede ser formal o informal. En el aprendizaje formal, los profesores organizan las actividades de aprendizaje, mientras que en el aprendizaje informal los estudiantes buscan su propio material en internet. El aprendizaje es a su propio ritmo, dependiendo de sus objetivos y ambición de aprender. Mientras que el aprendizaje electrónico se refiere a actividades de aprendizaje que involucran computadoras y redes interactivas, el aprendizaje basado en la web está asociado con materiales de aprendizaje que se entregan al estudiante a través de un navegador web.

Cualquier sitio web que contenga material de estudio en alguna área del conocimiento puede considerarse como ejemplo de aprendizaje basado en la web. La mayoría de estos sitios se construyen con base en páginas web que están ligadas por medio de hipervínculos y que contienen material de estudio en forma de textos, imágenes, menús, videos, etc. Dentro de este material se pueden incluir evaluaciones en forma de cuestionarios o exámenes con el objetivo de medir el nivel de aprendizaje del estudiante. Ejemplos de sitios de WBL lo constituyen tutoriales de alguna área de aprendizaje como medicina, matemáticas, idiomas como inglés o chino, etc. en forma de ambientes de aprendizaje. El concepto o término WBL hoy en día es menos usado y es considerado simplemente como *eLearning* u otros términos más modernos como “ambiente” o “plataforma de aprendizaje”. Además, si dichos ambientes personalizan el aprendizaje conforme al estilo de aprendizaje del estudiante, o a su capacidad académica o incluso a su estado afectivo, entonces se les nombra a estos “sistemas tutores inteligentes” o “ambientes de aprendizaje inteligente”.

2.3 Aprendizaje móvil

El aprendizaje móvil (*mobile-learning* o *mLearning*) también es una extensión del aprendizaje electrónico (*eLearning*), y se refiere al uso de dispositivos móviles, como asistentes personales digitales (PDA, por sus siglas en inglés), tabletas electrónicas, teléfonos celulares, teléfonos inteligentes, consolas de videojuegos portátiles, o cualquier otro dispositivo electrónico

portátil, en la educación. El aprendizaje móvil es un tipo de aprendizaje en el cual el estudiante no se encuentra en un lugar en específico. La ventaja más importante del *mLearning* frente al *eLearning* es el acceso por parte del alumno a la información demandada con independencia del tiempo y ambiente.

Existen muchas definiciones del término *mLearning*. Desmon Keegan (2000), responsable del proyecto *mLearning*, propuso una definición sencilla en términos de la capacidad para proveer educación y entrenamiento utilizando dispositivos electrónicos portables como PDA, *palmtops*, teléfonos celulares y *smartphones*. Lenher y Nosekabel (2002), por su parte, definieron el *mLearning* como “cualquier servicio o plataforma que provea al aprendiz con información electrónica general y contenido educacional que ayuden en la adquisición de conocimiento independientemente de la ubicación y el tiempo”.

En la figura 11 se puede apreciar que el *mLearning* es un subconjunto del *eLearning* debido a que ambos tipos de aprendizaje pueden usar computadoras para alojar los contenidos de aprendizaje. Sin embargo, en el caso de *mLearning* el dispositivo electrónico debe ser móvil. Por otra parte, el *eLearning* es un subconjunto de *dLearning* (aprendizaje a distancia) en el cual no se requiere que los estudiantes asistan físicamente a la escuela ni tampoco es indispensable contar con dispositivos tecnológicos, como computadoras, tabletas o teléfonos inteligentes.

El uso de dispositivos móviles se popularizó desde finales del siglo pasado principalmente para comunicación, pero para su uso en la educación, se requiere que los sistemas que se usen cumplan con ciertas características esenciales, las cuales han sido identificadas por Sharples *et al.* (2002) y Chen *et al.* (2002) en sus respectivos trabajos. Entre los requisitos que sugieren Sharples *et al.* para el *mLearning* se encuentran los siguientes:

- ✓ Aplicaciones de *software* con alta capacidad de movilidad.
- ✓ El *software* debe estar disponible (donde sea), en cualquier lugar, donde el usuario necesite aprender.
- ✓ El *software* debe ser adaptable, diseñado para apoyar el aprendizaje personalizado de acuerdo con el desarrollo de habilidades y conocimiento del usuario.

- ✓ El *software* debe ser persistente (acumulación de conocimiento), independiente de los cambios de tecnología.
- ✓ La interfaz debe ser amigable y de fácil usabilidad personal.

Desde una perspectiva pedagógica, el *mLearning* soporta una nueva dimensión en el proceso educativo. Las características del *mLearning* presentadas por Chen *et al.* (2002) son:

- ✓ Urgencia en la necesidad de aprendizaje.
- ✓ Iniciativa para adquisición de conocimiento.
- ✓ Movilidad en el escenario de aprendizaje.
- ✓ Interactividad del proceso de aprendizaje.
- ✓ “Dependencia del contexto” de las actividades instruccionales.
- ✓ Integración de contenido instruccional.

Al inicio de la década de 2000, el aprendizaje móvil presentaba desventajas relacionadas con las características de los dispositivos disponibles en ese momento, como:

- ☒ Tamaño limitado de pantallas para la interacción con el usuario.
- ☒ Dispositivos con baja capacidad de procesamiento.
- ☒ Dispositivos con limitada capacidad de memoria.
- ☒ Baja velocidad de procesamiento y acceso a memoria.
- ☒ Poca compatibilidad entre dispositivos, que afectaba el funcionamiento de las aplicaciones.

2.4 Aprendizaje híbrido

El término “curso híbrido”, usado a veces como sinónimo del término “*blended learning*”, se refiere a un modelo de diseño del curso que combina el tiempo de clase tradicional, cara a cara o presencial, con el trabajo en línea y fuera de clase. Se define también al aprendizaje híbrido como aquel en el que una parte (entre 25-50%) del tiempo de clase tradicional presencial se reemplaza con clase en línea o trabajo fuera de la clase. La sustitución de 25%

a 50% del tiempo en su clase con tiempo en línea y trabajo fuera de la clase es lo que diferencia los cursos híbridos de los cursos basados en la web, ya que en los híbridos los estudiantes continúan reuniéndose durante las horas normales de clase y utilizan el componente en línea solo para complementar tiempo cara a cara. Otra definición menciona que también se refiere a aplicar diferentes estrategias de aprendizaje para enseñar y aprender.

Durante la pandemia de COVID-19, algunas escuelas implementaron el modelo híbrido puesto que permitía que algunos estudiantes tomaran sus clases en línea usando algunas plataformas de videoconferencias como Zoom, Skype o Teams y otros tomaran sus clases en forma presencial, todos al mismo tiempo. Existe otro modelo parecido llamado “mixto” o “*blended*”. En este modelo, todos los estudiantes combinan clases presenciales en ciertos temas y para otros temas usan material de aprendizaje electrónico, móvil o de otro tipo. El modelo híbrido de aprendizaje ha demostrado ser muy importante para situaciones donde los estudiantes no tienen las condiciones para tomar sus clases 100% en forma presencial.

2.5 Sistemas administradores de aprendizaje

Un sistema administrador de aprendizaje (LMS por sus siglas en inglés) es un programa (*software*) que se emplea para administrar, distribuir y controlar las actividades de formación en *eLearning* (en cualquiera de sus diferentes formas como móvil, híbrido, etc.) de una institución u organización.

Los LMS ofrecen diferentes funciones para administrar usuarios, recursos, materiales y actividades de formación, además de controlar el acceso de los usuarios y realizar el seguimiento del proceso de aprendizaje, efectuar evaluaciones, generar informes, gestionar servicios de comunicación entre usuarios como foros de discusión, videoconferencias, entre otros.

Existe una gran cantidad de LMS, algunos muy populares como Moodle, Claroline y Blackboard, entre otros. Estos sistemas podrían clasificarse como de uso libre o comercial. Por ejemplo, Moodle y Claroline son plataformas de código abierto y *software* libre, mientras que Blackboard es comercial. A continuación, se describen brevemente estos sistemas.

Moodle

Moodle (moodle.org) es un sistema de *software* libre para la gestión de cursos, también llamado sistema administrador de aprendizaje (LMS) o “ambiente de aprendizaje virtual” (VLE por sus siglas en inglés). Moodle se ha vuelto muy popular entre educadores alrededor del mundo por ser una herramienta para crear sitios web dinámicos para los estudiantes. Moodle requiere ser instalado en algún lugar del servidor, en cualquiera de sus computadoras o en una compañía de alojamiento web.

Algunas características de Moodle son que permite escalar a desarrollos muy amplios, facilita cursos en línea completos, permite crear módulos de actividad (tales como foros, *wikis*, bases de datos etc.) para construir comunidades colaborativas de aprendizaje, permite la participación de alumnos en creación de glosarios, y facilita la creación de plataformas para cursos específicos, los cuales pueden abrirse a alumnos que se encuentren conectados en línea.

Blackboard

El sistema administrador de aprendizaje Blackboard (blackboard.com) contiene un conjunto de aplicaciones de *software* diseñadas para mejorar la enseñanza y el aprendizaje. Blackboard ayuda a los instructores a construir cursos en línea y a participar con estudiantes de forma interactiva gracias a una interfaz intuitiva y fácil de usar.

Claroline

Claroline (<https://www.claroline.com/#/home/accueil>) es una plataforma utilizada para aprendizaje (*e-learning*) y trabajo virtual (*e-working*). Esta plataforma es de código abierto y *software* libre, está disponible en múltiples idiomas y tiene una gran comunidad de desarrolladores y usuarios en todo el mundo. Esto permite que un instructor construya cursos en línea y gestione las actividades de aprendizaje y colaboración en la web. La plataforma Claroline ha sido desarrollada considerando la experiencia y necesidades de profesores e instructores en todo el mundo, y además, ofrece una interfaz intuitiva y clara de administración de espacios.

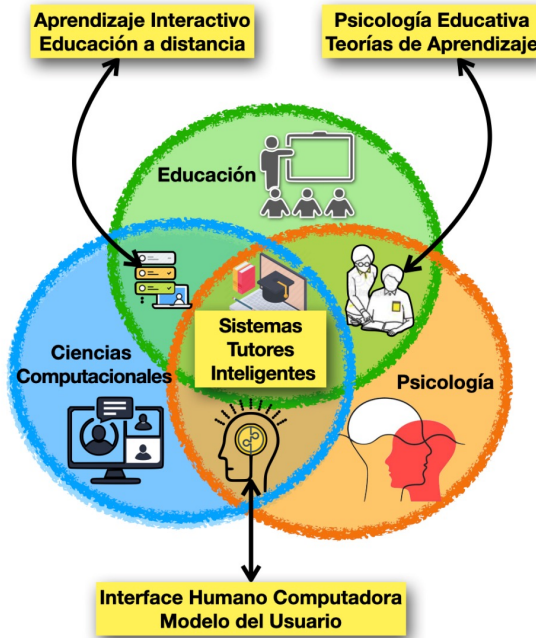
2.6 Sistemas tutores inteligentes

Con el desarrollo de la inteligencia artificial (IA), la evolución natural de los sistemas de aprendizaje como el *eLearning* son los sistemas tutores inteligentes (STI). Los STI comenzaron a desarrollarse en la década de los ochenta del siglo pasado con la idea de impartir el conocimiento utilizando alguna forma de inteligencia para asistir y guiar al estudiante en su proceso de aprendizaje. Por ejemplo, una de las ideas era simular el comportamiento de un tutor humano al adaptarse a las necesidades de cada estudiante, identificando para ello la forma en que éste resuelve un problema para poder brindarle ayuda cuando cometa errores.

Woolf (2009) define los STI como sistemas donde se modela la enseñanza, el aprendizaje, la comunicación y el dominio del conocimiento del experto con el entendimiento del estudiante sobre ese dominio. Por su parte, para VanLehn (1988) un STI es un sistema de *software* que utiliza técnicas de IA para representar el conocimiento e interactuar con los estudiantes. Para Polson y Richardson (2013), los STI son programas de computadora que incorporan técnicas de IA y pedagogía, y están diseñados para simular el comportamiento de un tutor humano. Para Nkambou *et al.* (2010), los STI tienen un objetivo común: proveer servicios de tutoría que soporten el aprendizaje. Por lo tanto, un STI es capaz de dar seguimiento al aprendizaje que ha construido el estudiante e identificar sus debilidades para elaborar una estrategia de enseñanza personalizada que se adapte al estilo cognitivo del estudiante.

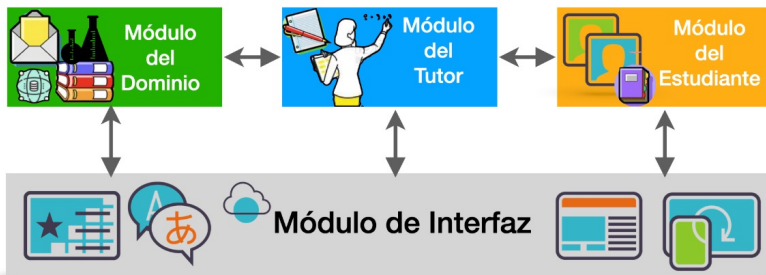
De acuerdo con Nwana (1990) y Woolf (2009), debido a que el proceso enseñanza-aprendizaje es un área tan compleja, los STI se auxilian de otras áreas del conocimiento como educación, psicología y ciencias computacionales, las cuales se complementan para desarrollar STI que sean eficientes y efectivos para la educación. En la figura 12, se aprecia la relación entre las diferentes áreas de conocimiento que se involucran para el desarrollo de sistemas tutores inteligentes. La intersección de cada área define subáreas de estudio que contribuyen a definir características importantes en los sistemas tutores inteligentes. Por ejemplo, en la intersección entre educación y ciencias computacionales se ubica el aprendizaje interactivo y la educación a distancia.

Figura 12. Áreas de conocimiento relacionadas con los sistemas tutores inteligentes



De acuerdo con la arquitectura propuesta por Woolf (2009), todo STI (véase figura 13) debe contener al menos un módulo de dominio donde se define el conocimiento del experto, un módulo de estudiante donde se representa el conocimiento del estudiante, un módulo de tutor que provee los mecanismos para el aprendizaje y la interfaz de usuario que permite que el estudiante interactúe con el STI.

Figura 13. Arquitectura típica de un sistema tutor inteligente



Módulo del tutor

El módulo del tutor cumple la función del profesor o tutor y es el encargado de generar los planes instruccionales de cada sesión. Este módulo define la estrategia pedagógica de enseñanza que el sistema aplicará con cada estudiante. También, contiene información sobre las tareas (materiales de aprendizaje) que se le presentan al estudiante de acuerdo con los objetivos de aprendizaje que el módulo del dominio establece y aquellos mecanismos para corregir el modelo del estudiante.

Módulo del dominio

El módulo del dominio representa todo el conocimiento del experto en los ámbitos relativos al proceso de enseñanza-aprendizaje y evaluación, y la metodología de enseñanza. Es el razonador que define dónde se almacenará la base de conocimiento y los mecanismos de resolución de problemas. Además, es el responsable de dirigir al módulo del tutor considerando los datos que se obtienen desde el módulo del estudiante.

Módulo del estudiante

El módulo del estudiante es el componente principal de un STI y tiene por objetivo realizar el diagnóstico cognitivo del alumno. Además, representa a quién se enseña (el estudiante), así como los estados cognitivos y afectivos del mismo, y registra su evolución a medida que avanza en el proceso de aprendizaje.

Módulo de interfaz

La interfaz de usuario es el mecanismo que permite la interacción entre el STI y el estudiante, su función principal es presentar el material del módulo del dominio y cualquier otro elemento didáctico (gráficas, ilustraciones, agentes inteligentes, entre otros) de la manera correcta. La interfaz de usuario es lo que permite a los estudiantes interactuar con el sistema.

2.7 Otros sistemas inteligentes de aprendizaje

Para Brusilovsky & Millán (2007) un “ambiente inteligente de aprendizaje” (ILE, por sus siglas en inglés) es la unión entre un STI y un entorno electrónico de aprendizaje que permite múltiples formas de navegación a través del dominio y un STI que dirige la enseñanza. El objetivo de los ILE es ofrecer al estudiante un entorno de aprendizaje que se adecue al estudiante, para lo cual utiliza técnicas de hipermedia que adaptan el contenido de los recursos hipermedia de manera dinámica; esta personalización sucede conforme a los gustos y preferencias del estudiante. Esta información se obtiene a través de la interacción del usuario con el sistema mediante un modelado del usuario, que es una representación individual de metas, preferencias, objetivos y conocimientos del estudiante. Para Mavrikis y Holmes (2019), los ambientes o entornos inteligentes de aprendizaje son una amplia categoría de aplicaciones educativas digitales interactivas que incluyen características que permiten proveer servicios personalizados y apoyo adaptativo a los estudiantes (ya sea mediante selección o adaptación de tareas, o asistencia mientras los estudiantes están realizando una tarea).

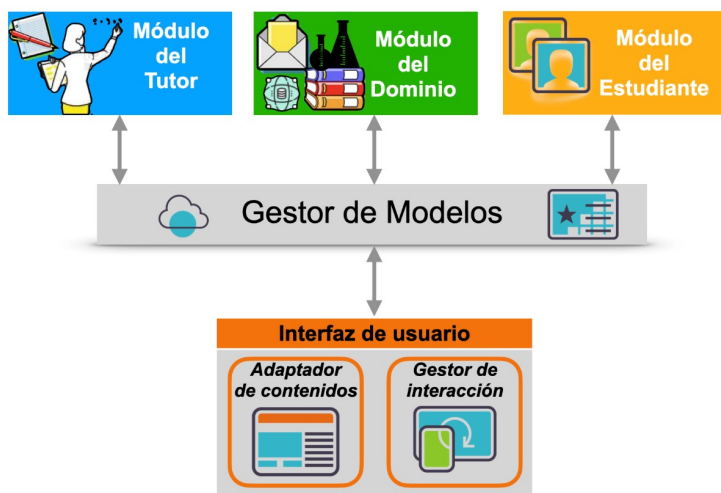
Por otro lado, los “ambientes afectivos de aprendizaje” funcionan de manera similar a un STI o ILE, con la diferencia de que dentro del modelo del estudiante se definen y guardan los estados afectivos del estudiante que interactúa con el sistema. Algunas características afectivas que se incorporan en el modelo del estudiante son las emociones y actitudes del estudiante, tales como la confusión, la frustración, la agitación, el aburrimiento, la motivación, la confianza en sí mismo y la fatiga.

Para Woolf (2009), la computación afectiva es el área encargada de analizar las emociones de la persona, por lo que muchas veces los investigadores se apoyan en los análisis de esta área para agregar detección de características afectivas en los ambientes de aprendizaje que consideran las emociones de los estudiantes. Pekrun *et al.* (2002) consideran que las emociones juegan un rol importante en el aprendizaje de los estudiantes. Por ejemplo, en un aula de clases tradicional, los alumnos experimentan toda clase de experiencias afectivas, tales como estar emocionados estudiando

para obtener una calificación satisfactoria, sentir orgullo por sus logros académicos, estar sorprendidos por un nuevo método de solución, experimentar ansiedad por fallar en un examen, sentir culpa por obtener calificaciones bajas en una o varias materias o sentir aburrimiento durante las clases. Todo esto lleva a que las emociones tengan un impacto real e importante sobre el aprendizaje del estudiante, pues afectan su rendimiento académico de manera positiva o negativa.

Por último, Luckin *et al.* (2016) definen a un “ambiente adaptable de aprendizaje” como “un ambiente digital de aprendizaje que adapta la enseñanza, los enfoques de aprendizaje y los materiales de acuerdo con las capacidades y necesidades del individuo que aprende” (p. 18). Esta adaptación se puede lograr agregando al módulo de interfaz de usuario de un STI la capacidad de analizar y usar la información que provee la interacción para la adaptación (figura 14).

Figura 14. Estructura de un ambiente adaptable de aprendizaje



3. Modelos de estilos de aprendizaje

Una definición ampliamente aceptada del término “estilos de aprendizaje” fue propuesta por Keefe & Thompson (1987), quienes sostienen que los estilos de aprendizaje son aquellos rasgos cognitivos, afectivos y fisiológicos que sirven como indicadores de la forma como los individuos perciben, interactúan y responden en los ambientes de aprendizaje.

Las teorías sobre estilos de aprendizaje se fundamentan en reconocer la individualidad de cada sujeto cuando aprende. Los ambientes de aprendizaje electrónico utilizan estas teorías para proveer materiales didácticos adecuados al estilo de aprendizaje de cada estudiante (Zatarain & Barrón, 2011). En distintos trabajos de investigación (Felder & Silverman, 1988; Cofield *et al.*, 2004; Gardner, 2000); los autores reconocen que los estudiantes presentan características propias que diferencian su forma de aprender de otros estudiantes. Cada estudiante se beneficia con cierto proceso de enseñanza, lo cual implica ofrecerle materiales didácticos e instruccionales diseñados específicamente para su configuración de estilo de aprendizaje.

Los modelos de estilos de aprendizaje categorizan los métodos de enseñanza utilizados por los profesores, así como las formas en que los estudiantes aprenden. El objetivo principal es que, en cada categoría que posea el modelo, las necesidades de aprendizaje de los estudiantes sean satisfechas (Felder, 1996). Pantoja *et al.* (2013) realizaron una recopilación de los modelos de estilos de aprendizaje y los clasificaron de acuerdo con sus características para establecer relaciones existentes entre los modelos.

A continuación, se presentan cuatro modelos que han sido de los más estudiados y aplicados en ambientes de aprendizaje electrónico.

3.1 Modelo de estilos de aprendizaje de Gregorc

El modelo de estilos de aprendizaje de Gregorc fue presentado por Anthony Gregorc y Kathleen Butler, y su objetivo es proveer una estructura organizada de la forma en que funciona la mente (Mills, 2002). Este modelo se basa en la existencia de percepciones, las cuales son la forma en que se evalúa el contexto. En este modelo, las percepciones del individuo se reconocen como la parte fundamental del estilo de aprendizaje.

En este modelo se reconocen cuatro estilos de aprendizaje:

- ✓ Secuencial concreto
- ✓ Aleatorio abstracto
- ✓ Secuencial abstracto
- ✓ Aleatorio concreto

Cada estilo de aprendizaje tiene diferentes fortalezas y debilidades, y se requieren distintas metodologías de enseñanza y materiales para cada uno.

La combinación de dos cualidades de percepción y dos habilidades de ordenamiento permiten obtener cada uno de estos estilos de aprendizaje. Las cualidades de percepción son abstracto y concreto, mientras que las habilidades de ordenamiento se tratan de aleatorio y secuencial. El modelo acepta que las cuatro características están presentes en todos los estudiantes, pero establece que una cualidad de percepción y una habilidad de ordenamiento son siempre más dominantes.

La cualidad de percepción concreta significa registrar la información a través de los cinco sentidos. Por otra parte, las percepciones abstractas se refieren a la comprensión de las ideas, cualidades y conceptos que no pueden ser vistos. La habilidad de ordenamiento secuencial involucra la organización de la información en forma lineal, mientras que la aleatoria organiza la información en bloques sin orden específico.

A los estudiantes de cada estilo de aprendizaje se les facilitan ciertas actividades. Las actividades que favorecen a cada estilo de aprendizaje son:

Secuencial concreto. Los estudiantes que tienen este estilo de aprendizaje trabajan dentro de los límites establecidos y fácilmente generan productos concretos a partir de ideas.

Secuencial abstracto. Los estudiantes con este estilo de aprendizaje prefieren obtener por adelantado toda la información, analizan las ideas, investigan para aprobar o rechazar teorías por medio de hechos.

Aleatorio concreto. Los estudiantes de este tipo generan ideas creativas y suelen asumir riesgos, son capaces de visualizar varias opciones y soluciones, además de inspirar a otros estudiantes a la acción.

Aleatorio abstracto. Estudiantes con este estilo de aprendizaje establecen buenas relaciones con todos los individuos y son buenos escuchando a sus compañeros, se enfocan en temas e ideas.

El modelo de Gregorc ha sido cuestionado respecto a la clasificación de estilos de aprendizaje que propone. Se señala que el estilo de aprendizaje de los estudiantes no tiene una relación clara con las categorías de su modelo. Incluso, Cofield *et al.* (2004) lo etiquetan como inadecuado para la identificación de los estilos de aprendizaje de los estudiantes, considerándolo teórica y psicométricamente incorrecto.

Los modelos de estilos de aprendizaje como el Gregorc están basados en su mayoría en cuatro modalidades: visual, auditivo, kinestésico y táctil. Otros modelos de estilos de aprendizaje que caen en la misma familia que el de Gregorc son los de Bartlett, Betts, Dunn Dunn, Gordon, Marks, Paivio, Richardson, Sheehan y Torrance (Cofield *et al.*, 2004).

3.2 Teoría de inteligencias múltiples de Howard Gardner

La teoría de Howard Gardner fue presentada por primera vez en el año de 1983, en el libro *Frames of Mind: The Theory of Multiple Intelligences* y desde entonces ha sido actualizada. Para Gardner cada estudiante está dotado de diferentes tipos de inteligencias —de ahí que su teoría sea conocida como

inteligencias múltiples (Gardner, 2000)— y estos se expresan en distintos niveles. Gardner además afirma que cada persona tiene un perfil cognitivo único.

En el año 2000 la lista de inteligencias de Gardner contaba con ocho diferentes tipos de inteligencias. Cada una de las inteligencias propuestas por Gardner fue evaluada y aceptada bajo la premisa de que un nuevo tipo de inteligencia debía servir para resolver problemas o dificultades.

Las primeras inteligencias identificadas por Gardner se presentan a continuación:

Inteligencia lingüística. Esta inteligencia está relacionada con la palabra hablada y escrita. Este tipo de inteligencia se refiere a la habilidad de un individuo para aprender idiomas y su capacidad para utilizar el lenguaje para alcanzar ciertos objetivos. Los estudiantes con este tipo de inteligencia aprenden mejor leyendo, tomando notas, escuchando clases, en discusiones y debates.

Inteligencia lógico-matemática. Esta inteligencia corresponde a los estudiantes que tienen la capacidad de analizar problemas lógicos, realizar operaciones e investigar problemas de manera científica. Tienen sus fortalezas en la lógica, abstracciones, razonamiento y el uso de números.

Inteligencia musical. Esta inteligencia está relacionada con el ritmo, la música y el proceso de escuchar. Los estudiantes con esta inteligencia tienen la habilidad de componer música y de utilizarla para aprender y memorizar otras cosas.

Inteligencia corporal-kinestésica. Los estudiantes con este tipo de inteligencia tienen facilidad para aprender cosas a través de sensaciones corporales. Además, suelen tener la capacidad de utilizar su cuerpo para resolver problemas.

Inteligencia espacial. Los estudiantes con esta inteligencia suelen tener buen sentido para ubicarse y orientarse en alguna dirección física. Esta inteligencia involucra la habilidad para reconocer el posible uso de patrones en espacios grandes.

Inteligencia interpersonal. Esta inteligencia permite a los estudiantes adaptarse fácilmente para trabajar con otras personas. Los estudiantes con esta inteligencia tienen la capacidad de comprender las motivaciones de otros individuos.

Inteligencia intrapersonal. En contraste con la inteligencia interpersonal, este tipo de inteligencia permite al estudiante comprenderse a sí mismo, es decir, comprender sus propias motivaciones.

Inteligencia naturalista. Esta inteligencia permite percibir las relaciones entre especies y clasificar organismos existentes en la naturaleza, además de entender los fenómenos naturales.

Desde la presentación de la teoría de inteligencias múltiples, Gardner ha evaluado la posibilidad de agregar otras inteligencias a su modelo; algunas inteligencias candidatas han sido la espiritual y la existencial. Los modelos de estilos de aprendizaje como el de Gardner contienen características muy sólidas de las estructuras cognitivas, incluyendo “patrones de habilidad”.

Otros modelos de estilos de aprendizaje que se agrupan en la misma familia que el modelo de Gardner son los de Broverman, Cooper, Guilford, Hulzman y Hudson, Hunt, Kagan, Kogan, Messick, Pettigrew y Witkin (Co-field *et al.*, 2004).

3.3 Indicador de tipo de Myers-Briggs

El modelo conocido como “indicador de tipo de Myers-Briggs” (MBTI por sus siglas en inglés) identifica cuatro escalas donde se ajustan todos los sujetos (Myers-Briggs, 2010). Las escalas son extraversión/introversión, sensitivo/intuitivo, razonamiento/sentimiento, y juicio/percepción.

Extraversión/Introversión (E/I). Esta escala separa a los estudiantes que son sociables y buscan estar con compañía, de aquellos estudiantes que prefieren estar a solas.

Sensitivo/Intuitivo (S/N). Esta escala identifica a los estudiantes que enfocan su atención en la realidad y a aquellos que le dan mayor importancia a su imaginación.

Razonamiento/Sentimiento (T/F). En esta escala se muestra la diferencia entre los estudiantes que se basan en la lógica y los que lo hacen en la intuición cuando se enfrentan a la toma de decisiones. A esta escala la describen las letras T y F por sus siglas en inglés (Thinking/Feelings).

Juicio/Percepción (J/P). Esta escala se refiere a los estudiantes que analizan y categorizan sus experiencias, y a aquellos que responden espontáneamente.

Estas cuatro escalas se combinan con el fin de obtener 16 tipos de personalidad. Cada una de las combinaciones de los tipos se describe con cuatro letras (una por cada escala). Por ejemplo, ENTJ representaría a la personalidad con características de extraversión, intuición, razonamiento y juicio.

Los modelos de estilos de aprendizaje como el de Myers-Briggs son considerados también de un tipo de personalidad relativamente estable. Existen otros modelos que caen en la misma clase o familia del de Myers-Briggs como los de Apter, Epstein y Meier, Harrison-Branson, Jackson y Miller (Cofield *et al.*, 2004).

3.4 Modelo de estilos de aprendizaje de Felder-Silverman

Richard Felder y Linda Silverman propusieron en 1988 un modelo de cuatro dimensiones o categorías, dos de las cuales tienen características que se encuentran en los modelos de Myers-Briggs y de Kolb. El modelo de estilos de aprendizaje de Felder-Silverman establece categorías para los estilos de aprendizaje de los estudiantes por medio de cuatro escalas que tienen dos dimensiones cada una.

Estas cuatro escalas son sensitivo/intuitivo, activo/reflexivo, visual/verbal y secuencial/global.

Sensitivo/Intuitivo. Esta escala caracteriza a los estudiantes que son prácticos, concretos, orientados hacia los hechos y los procedimientos, de aquellos que son más conceptuales, innovadores, orientados hacia las teorías.

Activo/Reflexivo. Esta escala separa a los estudiantes que aprenden manipulando las cosas y trabajando con otros, de aquellos que aprenden pensando acerca de las cosas y trabajando solos.

Visual/Verbal. Esta escala se refiere a los estudiantes que prefieren la presentación visual del material (tal como películas, cuadros o diagramas de flujo) y a aquellos que prefieren las explicaciones escritas o habladas.

Secuencial/Global. Esta escala distingue a los estudiantes que aprenden poco a poco en forma ordenada, y a los estudiantes que aprenden de forma holista.

En el modelo Felder-Silverman cada escala tiene dos polos o dimensiones que representan los extremos de cada escala. El modelo define 16 estilos de aprendizaje diferentes que se obtienen considerando las dos dimensiones y las cuatro escalas (2^4).

Las escalas de este modelo son continuas, es decir que establece que un estilo de aprendizaje puede tener una pertenencia alta, mediana o baja en cada escala, de modo que no hay una relación binaria de pertenencia a cada polo de las escalas.

Las características propias de cada estilo de aprendizaje resultan de la combinación de las características de cada una de las escalas. Por ejemplo, un estudiante puede tener un estilo de aprendizaje sensitivo-activo-visual-secuencial, con cada una de las dimensiones en distinto grado. Las actividades que facilitan el aprendizaje de un estudiante se generan con la combinación de las propiedades de cada una de las dimensiones que constituyen su estilo de aprendizaje.

Este modelo ofrece además un cuestionario que se utiliza para identificar las preferencias de los estudiantes respecto a cada una de las escalas (Felder & Soloman, 2004). Este cuestionario se conoce como Inventario de Estilos de Aprendizaje (ILSQ, por sus siglas en inglés). El cuestionario está compuesto por un total de 44 reactivos; cada escala del modelo tiene 11 preguntas asociadas. Las preguntas del cuestionario son de opción múltiple y cada una de las preguntas tiene dos posibles respuestas.

Para la evaluación del cuestionario se asigna un valor de -1 a cada pregunta cuya respuesta sea la opción *a*, y un valor de 1 a las que tengan como respuesta la opción *b*.

Como ejemplo del *Index of Learning Styles Questionnaire*,¹ a continuación se presentan las primeras cuatro preguntas traducidas al español.

¹ Disponible en <https://www.webtools.ncsu.edu/learningstyles/>

1. Entiendo mejor algo
 - a) si lo practico.
 - b) si pienso en ello.
2. Me considero
 - a) realista.
 - b) innovador.
3. Cuando pienso acerca de lo que hice ayer, es más probable que lo haga sobre la base de
 - a) una imagen.
 - b) palabras.
4. Tengo tendencia a
 - a) entender los detalles de un tema, pero no ver claramente su estructura completa.
 - b) entender la estructura completa pero no ver claramente los detalles.

La elección del modelo de estilos de aprendizaje a utilizar en la implementación de un ambiente de aprendizaje electrónico se basa en varios aspectos. El primero es el impacto; en este caso, el modelo de Felder-Silverman de estilos de aprendizaje es uno de los más populares e influyentes entre la comunidad de aprendizaje electrónico. Este modelo ha sido utilizado en el desarrollo de ambientes de aprendizaje inteligente, tales como CS388, TANGOW, LSAS, WHURLE, entre otros (Parvez & Blank, 2008; Stash *et al.*, 2004). El segundo es el instrumento que el modelo proporciona para identificar el estilo de aprendizaje de un estudiante. En este caso, el modelo de Felder-Silverman cuenta con el cuestionario de Felder-Soloman que ha sido validado y probado (Felder & Spurlin, 2005; Litzinger *et al.*, 2005; Zywino, 2003), lo cual le proporciona un sustento científico que la mayoría de los otros modelos carecen.

Otros modelos de la misma familia de Felder-Silverman son los de Herrmann, Kolb, Allison y Hayes, Honey y Mumford, Kaufmann, Kirton y McCarthy (Cofield *et al.*, 2004).

La tabla 1 presenta un resumen de las características y modelos de estilos de aprendizaje clasificados por Pantoja *et al.*, 2013.

Tabla 1. Modelos de estilos de aprendizaje

Autores/as	Categoría	Caracterización del aporte
Gregorc	Basado en la experiencia	Considera dos habilidades de mediación: percepción (concreto/ abstracto) y orden (secuencial/causal). La combinación de estas dos genera cuatro estilos de aprendizaje que son: concreto-secuencial, abstracto-secuencial, concreto-causal, abstracto-causal.
Gardner	Basado en los canales de percepción de información	Propone ocho tipos de inteligencias: lingüística, lógico-matemática, corporal–kinestésica, espacial, musical, interpersonal, intrapersonal y naturalista (esta última agregada posteriormente). Más la posible inclusión de otros tipos. Las categorías están influenciadas por factores internos y externos a cada persona.
Myers-Briggs	Personalidad	Desarrolla una prueba de personalidad con el fin de determinar la forma en que se percibe y utiliza la información. Se basa en una serie de dimensiones a las que se asocian rasgos bipolares: modo de percibir la información (extraversión/introversión), modo de utilizar la información (sensación/intuición), modo de tomar decisiones (razonamiento/ sentimiento) y postura frente al mundo (juicio/percepción).
Felder-Silverman	Basado en canales de percepción de información	Propone que el estilo de aprendizaje está compuesto por algún rasgo bipolar en cuatro dimensiones: sensitivo/intuitivo, activo/reflexivo, visual/verbal y secuencial/global.

En la tabla 2 se muestra otra clasificación de los modelos de estilos de aprendizaje realizada por Coffield *et al.* (2004).

Tabla 2. Clasificación de modelos de estilos de aprendizaje

Los estilos de aprendizaje están basados en cuatro modalidades: visual, auditivo, kinestésico y táctil.	Los estilos de aprendizaje contienen características sólidas de las estructuras cognitivas.	Los estilos de aprendizaje son el componente de un tipo de personalidad relativamente estable.	Los estilos de aprendizaje son preferencias de aprendizaje estables.
Gregorc Bartlett Betts Dunn Dunn Gordon	Gardner Broverman Cooper Guilford Hunt	Myers-Briggs Apter Epstein-Meier Harrison-Branson Jackson	Felder-Silverman Hermann Kolb Allison-Hayes Honey-Mumford Kaufmann

Fuente: Coffield *et al.* (2004).

4. Computación afectiva

Las emociones son sentimientos humanos asociados al estado de ánimo, temperamento, personalidad, disposición y motivación. Pero también, se relacionan con hormonas como la dopamina, noradrenalina y serotonina. La motivación conduce y dinamiza el comportamiento, mientras que la emoción proporciona el componente afectivo a la motivación, ya sea positiva o negativa.

Las emociones son elementos prominentes que siempre están presentes en la mente humana (Picard, 1995). Expresar emociones al comunicarse oralmente es una manera de complementar la información que el orador nos está dando. Esta información adicional complementa lo que está explícito en el mensaje lingüístico. Paul Ekman fue pionero en realizar estudios sobre los efectos de las emociones en el habla y sus relaciones con las expresiones faciales (Ekman, 1999). Sin embargo, en los últimos años, científicos de la computación han trabajado con el problema del reconocimiento automático de emociones, por medio de la clasificación de éstas usando técnicas de reconocimiento de patrones.

Conocer el estado emocional de un individuo permite recabar información relevante acerca de su estado psicológico; con esta información, un sistema tutor inteligente tiene más capacidad para interactuar con una persona y responder a sus necesidades. Los sistemas computacionales que realizan reconocimiento automático de las emociones pueden perfeccionar el desempeño, usabilidad y, en general, la calidad de interacción humano-com-

putadora para sus usuarios; por ejemplo, mejorar el aprendizaje de los estudiantes o el servicio de atención a clientes, entre otras aplicaciones.

Por otro lado, en las últimas décadas, los sistemas tutores inteligentes (STI) han incorporado la capacidad para reconocer y manejar el estado afectivo del estudiante, además de la identificación tradicional de su estado cognitivo. Estos STI cuentan con dispositivos y sensores especiales para medir y monitorear rasgos faciales, conductividad de la piel, tonos o características en la voz, entre otras señales, y con ello reconocer o identificar el estado emocional o afectivo del estudiante (Conati & Maclare, 2004; D'Mello *et al.*, 2007; Arroyo *et al.*, 2009). Sin embargo, esta área de investigación —la computación afectiva—, con menos de 30 años, aún se encuentra en desarrollo pues tradicionalmente los STI fueron concebidos y diseñados para medir estados cognitivos sin incluir el estado afectivo o emocional (Carbo-nell, 1970; Clancey, 1979; Anderson *et al.*, 1990; Aleven & Koedinger, 2002; Woolf, 2009).

La investigación en el área de computación afectiva incluye detección y respuesta a algún afecto. Mientras que en los sistemas que detectan afecto se identifican emociones como frustración, interés o aburrimiento, en los sistemas de respuesta a afectos se intenta transformar los estados emocionales negativos como la frustración o el aburrimiento en un estado emocional positivo como el de interés.

4.1 Definición de emoción y afecto

Una emoción, según la Real Academia Española (RAE, 2011) se define como “la alteración del ánimo en forma intensa y pasajera, agradable o penosa, y que va acompañada de cierta conmoción somática”. Por otra parte, la Asociación Americana de Psicología (APA por sus siglas en inglés) ofrece la definición de emoción como “un patrón de reacción complejo, que involucra elementos experienciales, conductuales y fisiológicos, mediante el cual un individuo intenta lidiar con un asunto o evento personalmente significativo” (APA, 2021).

Para distinguir emoción de afecto y sentimiento se ofrecen las siguientes definiciones (Stathopoulou & Tsihrintzis, 2010):

Afecto: Se define como un patrón de comportamiento observable el cual es la expresión de un estado de sentimiento o emoción subjetivamente experimentado (Association AP, 1984).

Sentimiento: Es una sensación que ha sido probada contra experiencias previas y ha sido etiquetada. Los sentimientos son aprendidos durante la vida de una persona, son personales y biográficos ya que cada persona tiene un conjunto diferente de sensaciones previas de las cuales atraer mientras se interpretan y se etiquetan los sentimientos (Deleuze & Guattari, 1988).

Emoción: Son consideradas como la proyección/manifestación de un sentimiento, y generalmente, una respuesta cognoscitiva a la información de la periferia. A diferencia de los sentimientos, la proyección de emociones puede ser genuina o fingida (Ekman, 1972).

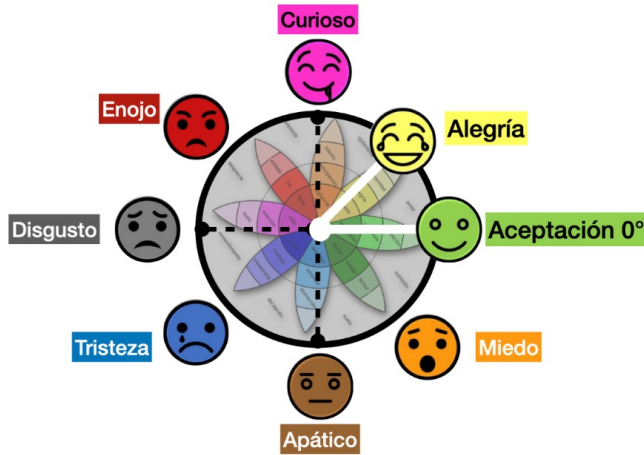
Algunas de las emociones básicas más comúnmente aceptadas son: ira, disgusto, miedo, felicidad, tristeza y sorpresa.

4.2 Categorización de las emociones

Existen muchas teorías psicológicas que tratan de explicar y clasificar a las emociones. De acuerdo con Wundt (2008), los estados emocionales se clasifican en tres dimensiones: 1) placer/codicia-desagrado/no-codicia, 2) emoción-tranquilidad, y 3) esfuerzo/tensión-relajamiento. Otra representación popular es el espacio “excitación-placer” que describe a la emoción como negativa, por ejemplo el enojo, o como positiva, como la felicidad.

Otra categorización de las emociones es lo que se conoce como “la rueda de las emociones” (Plutchik, 1980). En este caso, las emociones se distribuyen en un círculo y sus características se manejan como medidas de ángulos (figura 15). Las emociones en los extremos de cada eje van de lo negativo a lo positivo (por ejemplo, de apático a curioso).

Figura 15. Rueda de emociones



Fuente: Adaptación de Plutchik (1980).

Para el desarrollo de un sistema de *software* que reconozca emociones, resulta conveniente elegir un modelo con un número pequeño de emociones básicas. Ekman en su trabajo sobre análisis facial de expresiones (1999) describe un subconjunto de emociones que incluye sorpresa, miedo, alegría, enojo, disgusto/desprecio e interés, las cuales han sido utilizadas en sistemas tutores inteligentes que poseen capacidad para reconocer y tratar emociones y/o afectos (Arroyo *et al.*, 2009; D'Mello *et al.*, 2009; Forbes-Riley & Litman, 2009; D'Mello *et al.*, 2011).

4.3 Sistemas reconocedores de emociones en rostros

En esta sección se presentan algunas técnicas o métodos desarrollados para la implementación de sistemas para el reconocimiento automático de emociones a partir de rostros humanos.

Reconocimiento de caras

Para el reconocimiento de caras, un sistema computacional recibe una entrada, normalmente una imagen o video, y realiza un proceso para determi-

nar si en la imagen se encuentran caras humanas o no. Su estudio es un tópico muy importante en la biometría, donde se han hecho investigaciones al respecto (Zhao *et al.*, 2003; Ruiz-del-Solar & Navarrete, 2005; Delac & Grgic, 2007; Ponce & Karahoca, 2009).

Los términos “detección”, “identificación”, “autenticación” y “reconocimiento de caras” algunas veces se interpretan de forma similar, pero es importante señalar sus diferencias:

- a) La **detección** es el proceso para determinar si existe en una imagen o video un objeto que parezca una cara.
- b) La **identificación** es el proceso que permite identificar a qué persona pertenece la cara detectada en la imagen o video. Generalmente este proceso requiere un conjunto de imágenes o datos para identificar personas en específico.
- c) La **autenticación** es el proceso para verificar que la persona que está siendo detectada es en realidad ella y no otra.
- d) El **reconocimiento** viene a ser todo el proceso que involucra a los tres procesos de detección, identificación y autenticación.

De acuerdo con Chellapa, Wilson y Sirohey (1995), las investigaciones en reconocimiento facial de imágenes se dividen en dos categorías:

1. **Métodos holísticos:** estos métodos tratan a la imagen como un todo y el análisis se lleva a cabo sin intentar encontrar puntos individuales. La cara es reconocida como una sola entidad sin identificar regiones. Los métodos holísticos suelen usar técnicas de análisis estadístico, redes neuronales y transformaciones. Estos métodos por lo general requieren de una gran cantidad de datos de entrenamiento, y su ventaja es que, al usar toda la cara, no hay pérdida de datos por omisión, como sucede en otras técnicas. Sin embargo, solo son efectivos para reconocer caras desde una vista frontal.
2. **Métodos basados en características:** estos se basan en la identificación de ciertos puntos importantes de la cara, como los ojos, la nariz, la boca, etc. La localización de esos puntos puede ser determinada y usada para computar relaciones geométricas entre esos puntos. Por

lo tanto, el procesamiento se realiza de forma independiente para los ojos, la nariz, la boca y otros puntos importantes, y después se combinan para llevar a cabo el reconocimiento facial.

Técnicas de reconocimiento facial de emociones

Muchos estudios realizados para el reconocimiento de emociones en rostros humanos están basados en el trabajo de las seis emociones básicas propuestas por Ekman (1999). Esto se debe principalmente a que las emociones básicas representan propiedades universales entre personas de diferentes regiones, y además existe disponibilidad de diversos conjuntos de datos (*dataset*) digitales de rostros humanos o material relevante que se puede utilizar para el entrenamiento de sistemas automáticos de reconocimiento de emociones. Por otra parte, algunos investigadores también realizan esfuerzos para detectar estados afectivos no básicos usando expresiones faciales deliberadas que incluyen fatiga (Ji *et al.*, 2006), dolor (Littlewort *et al.*, 2007) y estados mentales como concentración, desacuerdo, interés, frustración e inseguridad (Kaliouby & Robinson, 2005). Los trabajos de investigación en el área de emociones y afecto se han orientado al análisis automático de datos de expresiones faciales espontáneas o no deliberadas, tal como se muestra en los trabajos de Cohn (2006), Bartlett *et al.* (2006), Littlewort *et al.* (2007).

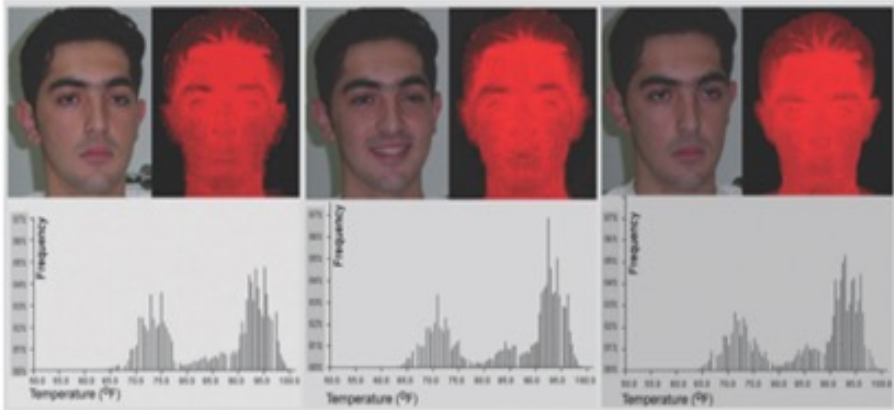
A continuación, se describen dos técnicas utilizadas para el reconocimiento de emociones en rostros humanos.

1. Reconocimiento de emociones basado en temperatura de la cara.

La temperatura corporal, y específicamente la temperatura del rostro, se usa para el reconocimiento de emociones. Los autores Khan, Ward, e Ingleby presentaron en 2009 un trabajo para reconocer emociones positivas y negativas utilizando la temperatura del rostro. La técnica consiste en captar imágenes del rostro y medir el calor corporal o temperatura de la cara cuando se presentan diferentes expresiones faciales. La figura 16 presenta un ejemplo de tres imágenes captadas de una misma persona con diferentes expresiones faciales, las cuales presentan, de izquierda a derecha, los estados emocionales neutral,

de felicidad y de tristeza. Las imágenes están acompañadas por la representación de calor corporal utilizando infrarrojo.

Figura 16. *Imágenes visibles e infrarrojas*



Fuente: Khan *et al.* (2009).

Después de una serie de pruebas, los autores definieron ciertos puntos de características termales faciales o *facial thermal feature points* (FTFP por sus siglas en inglés), los cuales tienen relación con los músculos del rostro y cambian su temperatura dependiendo del movimiento. La Figura 17 presenta a la izquierda el conjunto de puntos FTFP sobre la imagen de un rostro humano y a la derecha el conjunto de puntos FTFP sobre un mapa de músculos faciales.

Figura 17. *FTFP sobre cara humana y sobre mapa de los músculos faciales humanos*



Fuente: Khan *et al.* (2009).

2. **Reconocimiento de expresiones faciales utilizando AU.** Este método se basa en las unidades de acción (AU por sus siglas en inglés) que se determinan respecto a la contracción o relajación de ciertos músculos de la cara cuando se despliega una emoción. Ekman y Friesen (1978) desarrollaron el sistema de codificación de acciones de los músculos de la cara o “*facial action coding system*”, el cual actualizaron en 2002 (Ekman *et al.*, 2002) y contiene 46 AU para los músculos del rostro. En Valstar & Pantic (2006) se presenta un sistema que permite el reconocimiento de expresiones faciales de una forma fácil y robusta. El método de identificación se centra en detectar un subconjunto de 15 AU de las 46 AU definidas en el sistema de codificación de acciones de los músculos de la cara definido por Ekman y colegas. Valstar y Pantic solo consideraron las 15 AU más importantes a su criterio para reconocer emociones. Una de las ventajas de esta técnica es la facilidad de reconocer emociones espontáneas.

4.4 Sistemas reconocedores de emociones en voz

El reconocimiento automático de emociones a partir de la voz humana se fundamenta en el hecho de que los cambios en el sistema nervioso autónomo alteran indirectamente el habla, y esa información se aprovecha para producir sistemas capaces de reconocer emociones basados en la extracción de características en el habla. Por ejemplo, el habla o voz producida en un estado de temor, enojo o felicidad, se vuelve más rápida y más fuerte, precisamente enunciada con un rango de tono más alto y más ancho. El procesamiento de emociones en la voz reconoce el estado emocional de un usuario al analizar los patrones en esa voz. Para ello, los parámetros vocales y las características prosódicas, como las variables de tono y la velocidad de la voz, se analizan por medio de técnicas de reconocimiento de patrones (Dellaert *et al.*, 1996; Lee *et al.*, 2001).

Los sistemas de reconocimiento de emociones en el habla tradicionalmente se dividen en tres partes principales: procesamiento de la señal, extracción de características y clasificación.

Procesamiento de la señal

El procesamiento de la señal involucra la digitalización de la señal grabada, potencialmente un preprocesamiento de filtrado en la señal ya digitalizada, y por último la segmentación de la entrada de la señal en pequeñas unidades para el procesamiento posterior.

Extracción de características

El objetivo del cálculo o extracción de características es encontrar aquellas propiedades de la señal acústica previamente digitalizada y preprocesada que tengan las características necesarias para reconocer emociones.

La señal de habla puede ser asumida como una onda de sonido que es periódica por momentos. Como todas las ondas de sonido, esta tiene ciertas propiedades como amplitud, tiempo y frecuencia que sirven para caracterizarla y que ayudan a identificar las diferentes emociones. A continuación, se describen algunas de las medidas básicas de las características usadas para el reconocimiento automático de emociones usando la voz humana (Clark *et al.*, 2007).

1. **Pitch.** El término “pitch” se refiere a la percepción del oído respecto a la intensidad del tono. Esta es una propiedad muy obvia del habla, incluso para los que no son expertos, y a veces es considerada erróneamente como la más importante para la identificación de emociones.
2. **Formantes.** Los formantes son la máxima local en el espectro de frecuencia causados por la resonancia durante la producción del habla.
3. **Volumen.** El volumen es la fuerza del sonido tal como la percibe el oído humano. Es difícil medirlo directamente, por lo tanto, la energía de la señal a menudo es usada como una característica relacionada. La energía puede ser calculada desde el espectro después de aplicarle la transformada de Fourier a la señal original.
4. **Coeficientes “cepstrales” de frecuencia en escala de Mel (MFCC).** Son representaciones paramétricas de la señal de habla, muy comúnmente usadas en el reconocimiento automático del habla, y que se usan con gran éxito en la identificación y reconocimiento de emociones.

5. **Wavelets.** La transformada “wavelet” contiene muchos aspectos de la señal de habla, en particular información en la sincronización la cual es muy importante para el reconocimiento de emociones.
6. **Duración y tasa de habla o velocidad de habla.** La medida del tiempo juega un papel muy importante en la expresión de emociones. Esto implica la duración de las unidades de habla, como la longitud de la declaración, el promedio de longitud de una palabra en una declaración o el promedio de longitud de una sílaba dentro de una palabra.
7. **Calidad de voz.** La calidad de la voz puede sonar dificultosa, quebrada, áspera o en forma de susurro. Las relaciones de la calidad de voz con las emociones son diversas. Por ejemplo, las respiraciones dificultosas pueden resultar en que la emoción detectada sea de excitación.

Clasificación

El último paso es clasificar la emoción encontrada o identificada de acuerdo con las características extraídas en la voz. Una vez que han sido extraídas las características principales que son necesarias para reconocer emociones, se aplica alguna técnica de inteligencia artificial para la clasificación de la emoción. Algunas de las técnicas más populares para el reconocimiento son las redes neuronales artificiales, las máquinas de soporte vectorial (SVM por sus siglas en inglés), los árboles de decisión, los modelos ocultos de Markov y los algoritmos de votación. Cada una de estas técnicas tiene ventajas y desventajas. Sin embargo, la calidad del corpus o base de datos con voces que se utiliza para que el sistema reconocedor de emociones “aprenda” a clasificarlas, influye notablemente en la tasa de éxito que el reconocimiento alcanza.

4.5 Sistemas reconocedores multimodales

Un sistema multimodal para reconocer emociones combina dos o más entradas para percepción de los datos con el objetivo principal de incrementar la tasa de reconocimiento exitoso. Cuando se mencionan las entradas de percepción de los datos es posible realizar una semejanza con los sentidos (vista, oído, tacto, etc.) por medio de los cuales el ser humano percibe in-

formación que le provoca emociones. En un sistema multimodal de reconocimiento automático de emociones, la entrada puede ser una cámara web, un micrófono, un *mouse* o un sensor especial para detectar, por ejemplo, calor o sudoración en la piel.

Existen diversas técnicas de reconocimiento multimodal (Zeng *et al.*, 2008). El uso de sistemas multimodales para el reconocimiento de emociones fue inicialmente estudiado e implementado por Busso *et al.* (2004), quienes proponen un sistema bimodal para la identificación de emociones a través de la cámara y del audio de las expresiones. En Arroyo *et al.* (2009) y Woolf *et al.* (2007) los autores construyen un sistema multimodal para reconocer el estado afectivo de un grupo de alumnos. Los autores definieron cuatro estados afectivos, a saber: confianza, frustración, emoción e interés. El sistema reconocedor contiene cuatro dispositivos de entrada que se utilizan para captar las señales del usuario (véase figura 18). Las señales se captan a través de una cámara para medir o reconocer expresiones faciales, un brazalete inalámbrico que mide la conductividad en la piel, un asiento y respaldo sensibles a la presión corporal y un ratón de presión para medir la presión sobre su botón. Todo el *hardware* de los sensores fue construido en la Universidad Estatal de Arizona (ASU), con excepción de la cámara, que fue creada en el Instituto Tecnológico de Massachusetts (MIT).

Cada uno de los dispositivos capta las señales de entrada, las cuales se procesan y como resultado se determina el estado afectivo del usuario, que puede ser confianza, frustración, emoción o interés.

Figura 18. Las cuatro interfaces de entrada



Segunda Parte

APLICACIONES Y CASOS PRÁCTICOS

5. EDUCA: una herramienta de autor para cursos de aprendizaje electrónico y móvil

El diseño de materiales educativos para cursos móviles y a distancia, con diferentes estilos de aprendizaje, representa uno de los mayores desafíos. La mayor parte del trabajo actual en sistemas de aprendizaje para dispositivos móviles y no móviles está basado en el modelo de un autor (normalmente un profesor), quien construye los cursos, y muchos usuarios o estudiantes, los cuales reciben o se inscriben a esos cursos. Este modelo o paradigma es muy limitado y puede mejorarse al hacer uso de los avances tecnológicos, como es el caso del internet y en particular lo que se denomina la web 2.0, en donde los usuarios de las aplicaciones se convierten en los autores del propio conocimiento construido con las mismas. Hoy en día la información y conocimiento que mantienen muchas aplicaciones como Wikipedia, YouTube, Digg y muchas más, se genera por los mismos usuarios. Este enfoque colaborativo puede trasladarse también al campo educativo con el objetivo de que la construcción de los diferentes materiales didácticos lo lleven a cabo no solo los autores tradicionales de ello, los profesores o tutores, sino los mismos usuarios o estudiantes. Una razón importante es el hecho de que los estudiantes tienen una gran capacidad para buscar y encontrar nuevos recursos educativos cuando sienten la necesidad de ampliar o comprender mejor los temas contemplados en los cursos.

5.1 ¿Qué es EDUCA?

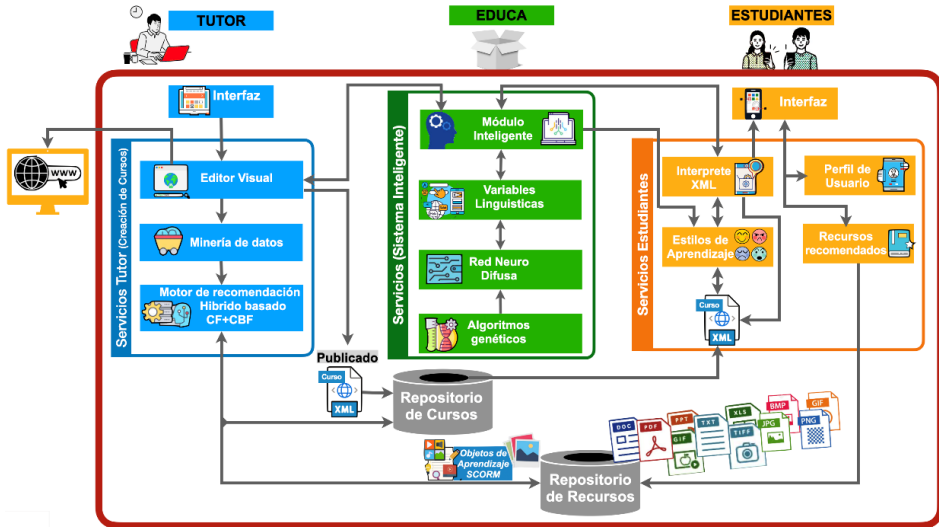
EDUCA (Zatarain Cabada *et al.*, 2011) es una plataforma de *software* que facilita la creación de material de aprendizaje adaptativo dentro de un ambiente colaborativo web 2.0, el cual puede ser utilizado en dispositivos móviles y en plataformas para administrar el aprendizaje.

La herramienta proporciona un sistema de autor el cual utiliza un sistema de recomendación para filtrar el material que los usuarios quieren agregar para ser parte de un curso. Además, usa la técnica de minería de datos para la búsqueda de material en la web de acuerdo con etiquetas previamente definidas. Por otra parte, los cursos generados por la herramienta se adaptan al mejor estilo de aprendizaje del estudiante bajo el modelo de estilos de aprendizaje de Richard Felder y Linda Silverman (1988). Para la implementación de los cursos adaptativos, los cuales también se pueden considerar tutores inteligentes, se usó una red neuronal artificial de tipo neurodifusa con apoyo de un algoritmo genético.

5.2 Arquitectura de EDUCA

La figura 19 ilustra la arquitectura principal de EDUCA. Como se puede observar en la figura, el sistema contiene tres módulos principales —a saber: Servicios Tutor (Creación de cursos), Servicios (Sistema inteligente) y Servicios Estudiantes—, existen dos actores principales: el tutor y los estudiantes. Un tutor inicialmente crea un curso o sistema tutor inteligente (STI) usando un editor visual. El curso puede crearse ya sea importando archivos ya preparados en formatos .html, .pdf y/o .doc, o bien, importando objetos de aprendizaje en formato SCORM (2008) desde cualquier repositorio de objetos. El autor también puede escribir material didáctico directamente con el editor visual.

Figura 19. Arquitectura principal de EDUCA



El autor (tutor) puede agregar materiales de aprendizaje a un curso y posteriormente el sistema genera cuatro diferentes instancias del curso correspondientes a los cuatro estilos de aprendizaje (activo-reflexivo, sensorial-intuitivo, visual-verbal y secuencial-global) del modelo de Felder y Silverman. Existe una interfaz especial en el **Módulo Inteligente** para ayudar al autor a construir el curso. El **Módulo Inteligente** contiene una interfaz gráfica gracias a la cual el usuario edita las funciones de membresía difusas.

El sistema tiene siete variables difusas de entrada:

- Orden de selección de respuestas.
- Número de aciertos (respuestas correctas).
- Tiempo (segundos) para responder los cuestionarios.
- Tiempo (segundos) para visitar los temas.
- Número de intentos hasta responder correctamente una pregunta.
- Número de veces que se accedió a una pregunta.
- Número de veces que se accedió a un tema.

Los valores de salida difusos se ingresan a una red neurodifusa, la cual ha sido entrenada previamente con valores reales. La salida de la red es el estilo de aprendizaje del estudiante.

Con la finalidad de optimizar el conjunto de pesos que intervienen en la red neurodifusa, se implementó un algoritmo genético. Los pesos se ordenan utilizando el algoritmo Bucket Sort en tres etapas: primero por capas, segundo por la neurona con arista de salida y finalmente por la neurona a la cual llega la arista. De esta forma se obtiene un cromosoma cuyos genes son números reales. La red artificial fue entrenada para 800 generaciones, con una población de 150 cromosomas.

La salida del editor visual de EDUCA es un curso de tipo sistema tutor inteligente para dispositivo móvil (MITS por sus siglas en inglés) en formato XML o un archivo SCORM. El curso en formato SCORM puede ser visualizado en cualquier sistema administrador de cursos (LMS) compatible con archivos SCORM. El curso también puede exportarse a un dispositivo móvil, incluyendo un intérprete XML. El intérprete contiene la red neuronal y su función es desplegar el material del curso en el dispositivo móvil, de acuerdo con el estilo de aprendizaje actual del estudiante o usuario del curso.

El estudiante es el otro actor en el sistema. Cuando el estudiante accede a un curso por primera vez el sistema genera su perfil de usuario. Este perfil se actualiza dinámicamente mientras el usuario interactúa con el sistema para acceder al contenido del curso. El perfil de estudiante contiene información relevante como:

- El estilo de aprendizaje detectado
- Información de evaluaciones
- Recursos recomendados en EDUCA
- Nivel de estudios del usuario (licenciatura/maestría/doctorado)
- Promedio general actual.

Una vez que un curso ha sido creado, el módulo Editor Visual lo guarda en el repositorio de cursos. Los cursos almacenados estáticamente están disponibles para ser accedidos por los estudiantes, los cuales, por razones obvias, consultan otros recursos o referencias en diferentes sitios web. Por ejemplo, un estudiante, después de estudiar un tema de un curso de Com-

piladores almacenado en el repositorio, siente la necesidad de reforzar el tema de Árboles Sintácticos Abstractos y por lo tanto decide buscar más información en la web. Una vez que el estudiante encuentre material de su agrado, él mismo puede recomendar esos recursos para ser incorporados al curso de Compiladores. EDUCA cuenta, en ese sentido, con un sistema inteligente de recomendación de recursos, los cuales son almacenados en un repositorio. Por otro lado, EDUCA asiste al estudiante con un módulo de minería de datos para la búsqueda de recursos en la web.

5.3 Creación de cursos en PC y visualización en dispositivos móviles

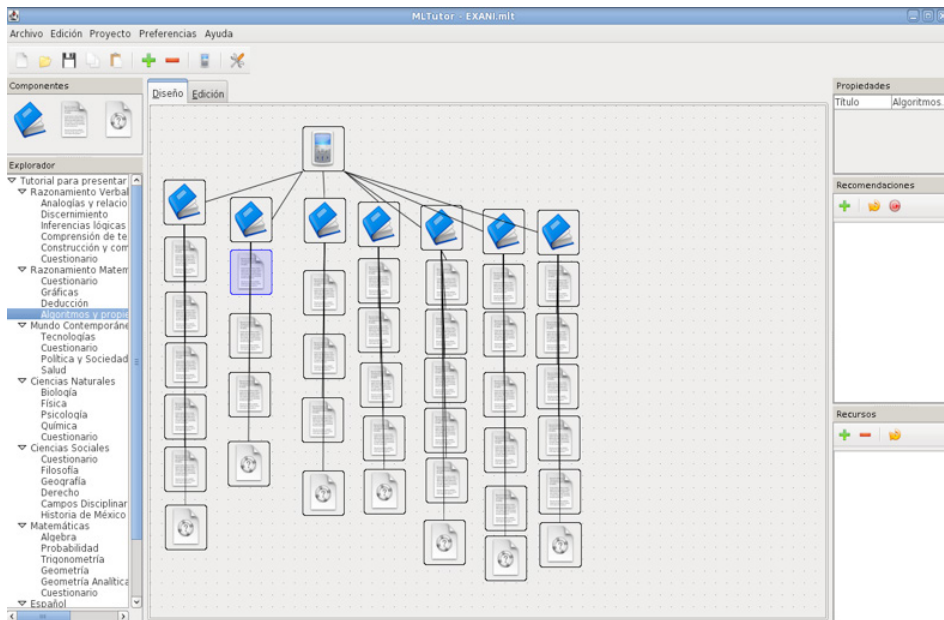
La aplicación de escritorio integrada en EDUCA permite al usuario crear cursos que se desplegarán en un dispositivo móvil o que serán guardados en formato SCORM. Al inicio, EDUCA muestra una interfaz de bienvenida (figura 20) con operaciones de usuario habituales. Como se observa en la figura 20, el nombre de la aplicación del *software* es “MLTutor”, que significa *Mobile Learning Tutor*.

Figura 20. Interfaz de bienvenida al usuario



En la figura 21, se presenta como ejemplo un curso de preparación para el examen de admisión del CENEVAL a los institutos tecnológicos federales (EXANI). En la creación del contenido temático del curso, sólo hay que utilizar las herramientas de edición que la aplicación proporciona. En esta parte de la creación del curso es donde surge el sistema de recomendación con soporte en web 2.0 y la utilización de minería de texto. El sistema de recomendación aparece como una lista de recursos sugeridos para ser agregados al tema actualmente en edición. Si el usuario los considera relevantes y de ayuda para complementar el contenido del tema, los agrega y entonces aparecerán en la lista del usuario como ayuda adicional para la comprensión del tema.

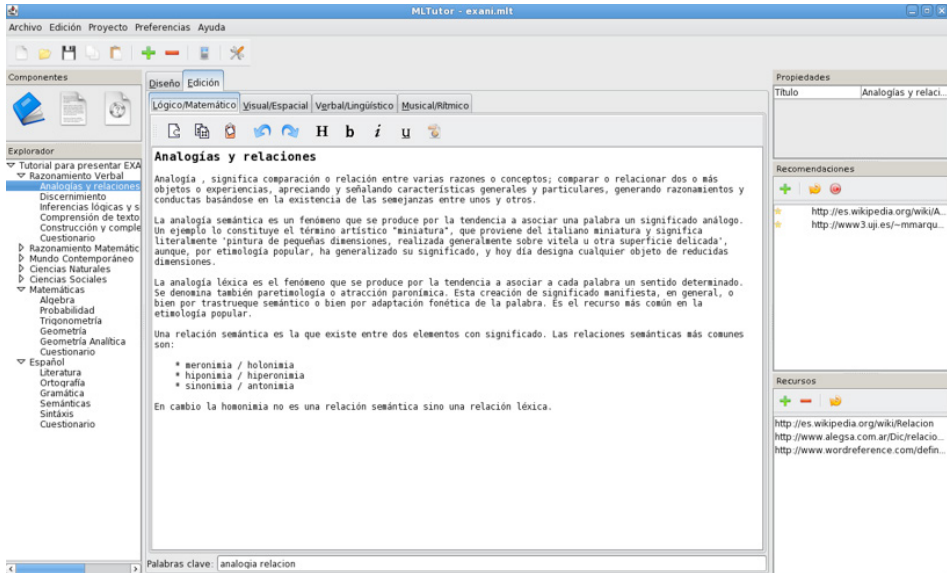
Figura 21. Estructura de un curso creado en EDUCA



El sistema de recomendación despliega la lista de recursos basándose en las palabras clave contenidas en el tema. Estas palabras pueden ser especificadas por el usuario que crea el tema, o pueden ser extraídas por medio de la herramienta de minería de texto integrada en la aplicación, la cual sugiere una serie de palabras clave que pueden resumir el contenido temá-

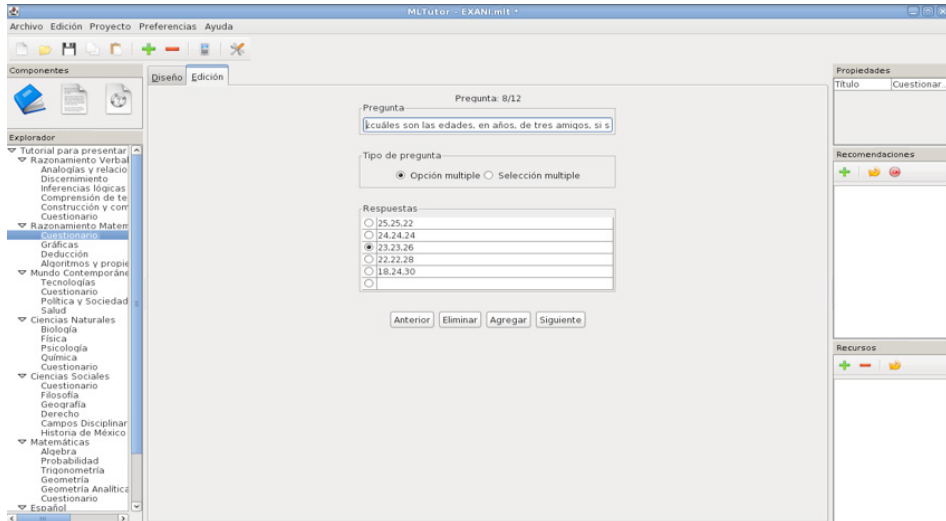
tico en edición. En la figura 22, se muestra la edición de un tema de EXANI, donde puede observarse el despliegue de la recomendación de distintos recursos (parte derecha de la interfaz), y la especificación de las palabras clave que describen al tema (parte inferior).

Figura 22. Edición de un tema con recomendaciones



Ya que la interacción del usuario es primordial para la adaptación de los cursos a su tipo de aprendizaje, es necesaria la creación de cuestionarios que maximicen dicha interacción. En EDUCA es posible tener dos tipos de preguntas: de opción única o de opción múltiple. La figura 23 muestra lo sencilla que es la interfaz para la creación de dichos cuestionarios.

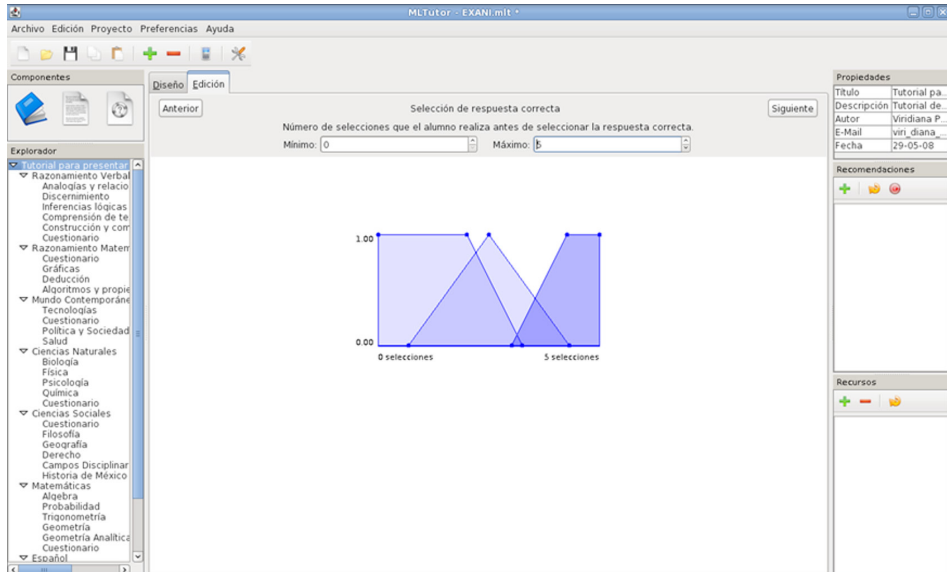
Figura 23. Creación de cuestionarios en EDUCA



Al final, es necesario especificar la definición de los conjuntos difusos que actúan sobre las variables de interacción con el usuario final. Dicha especificación se realiza con la definición de los límites inferior y superior de tales variables lingüísticas, además de los rangos de membrecía para los conjuntos difusos. Esto último se logra mediante sencillas operaciones de arrastrar y soltar figuras triangulares. Recuerde que dichos conjuntos difusos definen para el curso los valores de Bajo, Normal o Alto. La figura 24 muestra la especificación de la variable Selección de respuesta correcta en el curso EXANI.

Como puede observarse, la creación de un curso es una operación sencilla que permite al usuario profundizar en las tareas de definición de material del curso, y no en su creación. La utilización de la web 2.0 en el sistema de recomendación y de técnicas de inteligencia artificial como la minería de texto, van implícitas en la creación de cursos, facilitando su uso al usuario.

Figura 24. Especificación de variable lingüística



5.4 Motor de identificación de estilos de aprendizaje

La instalación de EDUCA en un dispositivo móvil contiene tres elementos (curso o tutor, red neuronal e intérprete del curso), y además se almacena el perfil del estudiante. El curso contiene todo el contenido agregado por el autor durante el proceso de creación del curso. El estudiante interactúa con el sistema (curso) a través de dos acciones principales: navegar en los temas del curso y responder los cuestionarios. La información generada durante la interacción entre el tutor y el estudiante se utiliza para actualizar dinámicamente el estilo de aprendizaje del estudiante y su perfil.

En la figura 25 se muestra un esquema de cómo el estudiante accede a las evaluaciones del curso y sus respuestas se utilizan para actualizar su perfil. La interacción entre el usuario y el sistema se da a través de menús de navegación que le permiten acceder a los capítulos, temas y contenidos, así como a los cuestionarios. Cuando el usuario intenta visualizar un tema, el intérprete toma la información del estilo de aprendizaje del estudiante y

selecciona los contenidos más adecuados para él. El **Intérprete** (un “*parser*” del curso empacado en formato XML) obtiene el estilo de aprendizaje del estudiante desde el componente **Perfil Estudiante**. El perfil del estudiante contiene la información del estudiante que se genera durante su interacción con el sistema, los resultados de las evaluaciones que el usuario realiza en el curso y su estilo de aprendizaje actual. El sistema contiene un módulo de **Identificación de estilo de aprendizaje** que se usa para determinar el estilo de aprendizaje del estudiante. Este módulo fue implementado a través de la red neuronal artificial.

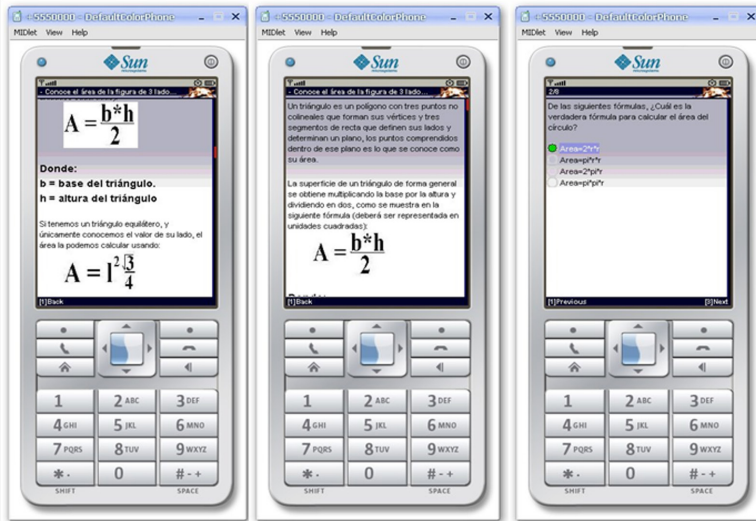
Figura 25. Identificación de estilos de aprendizaje



La red neuronal recibe dos entradas: la primera es el estilo de aprendizaje actual que usa el sistema para desplegar el contenido del material didáctico (el estilo de aprendizaje actual del usuario), y la segunda es el rendimiento del estudiante durante la sesión (resultados de las evaluaciones que se le han aplicado). La red neuronal proporciona como salida el nuevo estilo de aprendizaje del estudiante.

La figura 26 muestra la visualización de un curso en un simulador de dispositivo móvil. Las imágenes izquierda y central muestran cómo el dispositivo móvil despliega parte del contenido de un curso de matemáticas, mientras que la imagen a la derecha presenta un tercer dispositivo móvil con una pregunta y sus múltiples opciones de respuesta.

Figura 26. Visualización en simulador de un curso de matemáticas

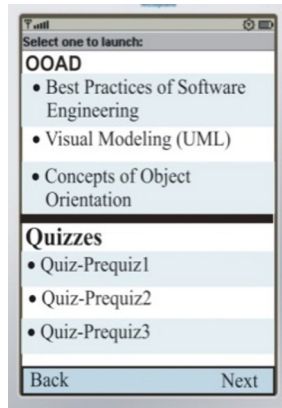


En la década pasada, los dispositivos móviles tenían muchas limitaciones en cuanto a sus características: tamaño de la pantalla, velocidad de procesamiento, memoria limitada, etc. El despliegue de la información de un curso en un dispositivo móvil representó una desventaja debido a las limitaciones de tamaño de pantalla de algunos dispositivos. Algunos autores que abordan el problema de usabilidad en dispositivos móviles consideran que los problemas de usabilidad principales son los siguientes:

- Acomodo de la información
- Accesibilidad
- Tamaño de la pantalla
- Navegación
- Duración de la batería
- Eficiencia.

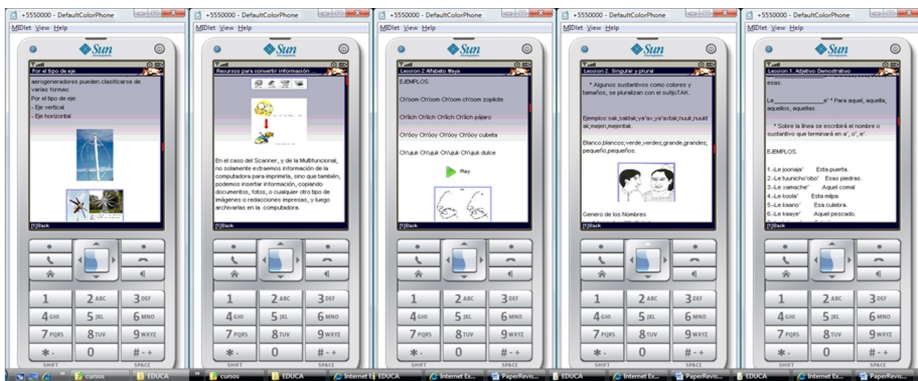
Siguiendo las normas encontradas en la información, se procedió a la creación de un nuevo conjunto de interfaces para poder visualizar un curso en un dispositivo móvil, con el fin de incrementar la capacidad de lectura para cualquier usuario. La figura 27 ilustra un ejemplo del despliegue de una interfaz en el simulador de un móvil.

Figura 27. Despliegado de Cursos en dispositivo móviles (simulación)



Los autores pueden usar una computadora de escritorio para crear los cursos. El curso puede incluir diferentes componentes, como textos, imágenes y cuestionarios, así como otras características de acuerdo con el tipo de inteligencia del estudiante como se muestra en la figura 28. El estudiante además podrá visualizar dicho curso desde diferentes tipos de dispositivos móviles. Esto agrega otro problema pues existe una gran cantidad de equipos, donde la variedad de las características de cada una de ellas —como memoria, pantalla, teclado, sistema operativo, etc.— es muy alta. Unas de las ventajas de EDUCA es que fue implementado en el lenguaje Java y que los cursos son guardados en formato XML. Esto le da un soporte de compatibilidad en diferentes equipos que soportan la plataforma Java.

Figura 28. Despliegado de diferentes componentes



5.5 Diseño de la red neuronal para clasificación de estilos de aprendizaje

La identificación de estilos de aprendizaje es una práctica que se ha llevado a cabo en los últimos años a raíz del reconocimiento de la individualidad de los estudiantes en sus procesos de aprendizaje. Para esto se emplean distintos métodos, los cuales involucran, generalmente, el uso de herramientas que cuestionan a los sujetos acerca de sus preferencias de estudio. Este tipo de métodos para la identificación del estilo de aprendizaje presentan tres grandes desventajas. La primera se refiere a que el estudiante toma un buen tiempo en responder las preguntas del cuestionario. El segundo inconveniente es el grado de confianza que se tiene en las preguntas del cuestionario para la identificación del estilo de aprendizaje. Factores que pueden influir en la disminución de la certidumbre de las respuestas pueden ser: 1) que la pregunta sea mal interpretada por el estudiante, 2) que el mismo no pueda elegir una respuesta, de entre las posibles, con la que sienta mayor afinidad o 3) que simplemente no se responda con la seriedad requerida. Una tercera desventaja es el hecho de que la mayoría de los sujetos no poseen una configuración de estilos de aprendizaje estática, sino que estos presentan cierto grado de cambio dependiendo de múltiples factores (estado de ánimo, lugar en el que se encuentre, tema estudiado, etc.). Al reconocer que los estudiantes poseen diferentes estilos de aprendizaje, es necesario que se generen, en consecuencia, cursos específicos para cada estudiante de acuerdo con su estilo de aprendizaje. El módulo inteligente toma como entrada un curso creado para los diferentes estilos de aprendizaje y una red neurodifusa para clasificar el estilo de aprendizaje del estudiante o usuario con el fin de producir como salida un curso o tutor inteligente.

Un curso se forma de dos componentes: un archivo XML, el cual almacena la información y datos del curso, y un intérprete que despliega la información del curso en el móvil. Dentro del intérprete va incorporada la red neurodifusa que ayudará a adaptar dinámicamente el curso al estilo de aprendizaje del estudiante o usuario.

Uso de una red neurodifusa

Para el diseño e implementación de la red neuronal de EDUCA se eligió un modelo híbrido neurodifuso. Este consiste en una red neuronal que recibe valores duros y, por medio de operaciones de conjuntos difusos, genera una salida, como si se tratase de cualquier otro tipo de red (Zatarain-Cabada *et al.*, 2009). Los sistemas neurodifusos combinan la computación en paralelo y las técnicas de aprendizaje de las redes neuronales, con la representación del conocimiento humano de la lógica difusa. Su funcionalidad es equivalente a un sistema de inferencia difuso.

La red puede ser entrenada para desarrollar reglas difusas del tipo IF-THEN y determinar los grados de membrecía para las entradas y salidas del sistema. La red neuronal contiene cinco capas, lo cual es casi una convención cuando se habla de sistemas híbridos neurodifusos.

Uso de una red de Kohonen

Uno de los problemas que surgieron al implementar la red neuronal neurodifusa fue el de obtener valores correctos para la salida deseada que se utilizan para entrenar la red neuronal. Ello requiere que grupos de expertos definan cuáles son las salidas deseadas para determinadas entradas. Por ejemplo, cuál es el estilo de aprendizaje de un estudiante que obtiene una calificación x en determinadas preguntas o ejercicios, ante un número y de errores y un determinado tiempo de respuesta. Ante lo complejo del problema, se decidió usar una red neuronal de Kohonen, la cual no necesita para su entrenamiento contar con salidas deseadas y, por lo tanto, no requiere de expertos.

Un mapa autoorganizado de características (o red neuronal de Kohonen) se entrena con aprendizaje no supervisado o competitivo (Negnevitsky, 2005). En el aprendizaje competitivo, las neuronas compiten entre sí para ser activadas. El modelo de Kohonen proporciona un mapeo topológico de un número fijo de patrones de la capa de entrada a una salida de mayor dimensión llamada “capa de Kohonen” (figura 29). La capa de Kohonen consta de una única capa de neuronas de cálculo, con dos tipos de conexiones diferentes: conexiones hacia adelante (desde la capa de entrada a la capa de Kohonen) y conexiones laterales entre las neuronas en la capa de Koho-

nen. Durante el entrenamiento, la red neuronal recibe nuevos patrones de entrada, los cuales se representan en forma de vectores. Cada neurona en la capa de Kohonen recibe una copia modificada del patrón de entrada. Las conexiones laterales producen efectos excitadores o inhibidores, en función de su distancia respecto a la neurona ganadora.

5.6 Entrenamiento de la red neuronal

Se implementó una red de Kohonen con 1600 neuronas formadas en un arreglo de 40x40. La salida de la red consiste en tres señales de valores en el rango de -1 a +1, los cuales se convierten en el rango de -11 a +11, que es el que se utiliza en el cuestionario llamado *Index of Learning Styles Questionnaire* (ISLQ) de Felder y Soloman (2004) y que representa el estilo de aprendizaje del estudiante.

El espacio de datos utilizados para el entrenamiento de la red fue recolectado a través de un proceso que involucró trabajo de campo. En este trabajo se seleccionó una muestra de 47 sujetos para pruebas, todos ellos estudiantes de educación media superior. A cada estudiante se le proporcionó el Cuestionario del Inventario de Estilos de Aprendizaje de Felder-Soloman y sus respuestas fueron almacenadas. A través de este cuestionario se identificó y registró el estilo de aprendizaje de cada estudiante que participó en el estudio. Se elaboró material didáctico de tres temas distintos: fotografía, energía eólica y computación para crear cursos de cada tema. De cada uno de estos cursos se crearon ocho versiones distintas, cada una para cada estilo de aprendizaje distinto. Dado que solo se toman en cuenta tres dimensiones de los estilos de aprendizaje de Felder-Silverman, en total solo existen ocho estilos de aprendizaje diferentes. Los ocho estilos de aprendizaje distintos son resultado de la combinación de las dimensiones visual-verbales, sensitivo-intuitivas y secuencial-globales.

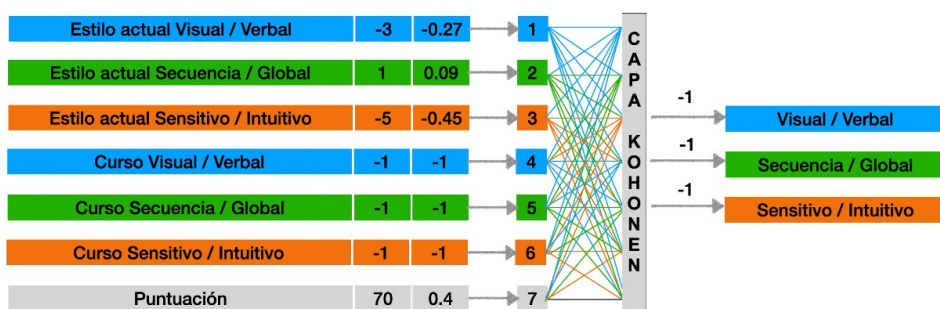
A cada estudiante que participó en el ensayo le fue asignada solo una versión de cada curso, la cual se asignó aleatoriamente. A todos los estudiantes se les permitió estudiar el material de los cursos por un tiempo aproximado de 40 minutos. La lectura del material requiere de 15 minutos aproximadamente, de tal manera que el estudiante podía dedicar el

tiempo restante a analizar información o realizar tareas auxiliares que le facilitaran la comprensión del tema. Al finalizar el tiempo de estudio, los estudiantes dedicaron otros 15 minutos para responder una evaluación relacionada con el tema del curso. Las evaluaciones consistieron en una serie de preguntas con respuestas de opción múltiple. Todos los estudiantes respondieron la misma evaluación por cada tema. Los estudiantes dedicaron un total de 55 minutos en estudiar el material y realizar la evaluación. Cada estudiante participó en tres sesiones de 55 minutos, una por cada curso.

En la figura 29 se muestra una representación gráfica del entrenamiento de la red neuronal. Como se observa en la figura, las primeras tres neuronas reciben como entrada tres valores correspondientes al estilo de aprendizaje del estudiante obtenido por medio del cuestionario ISLQ aplicado a cada uno de los estudiantes de la muestra. Los siguientes tres valores se obtienen de los tres estilos de los tres cursos en que participó cada estudiante. Por último, la puntuación, que es la calificación obtenida por el estudiante. La salida en el ejemplo (figura 29) muestra los valores para cada estilo de aprendizaje del estudiante.

Cabe mencionar que el estilo activo-reflexivo no se incluyó como salida en la red neuronal debido a que se consideró que los estudiantes con este estilo, no son candidatos para utilizar móviles para el aprendizaje.

Figura 29. Red neuronal durante el entrenamiento (datos hipotéticos)



La naturaleza de los ambientes de aprendizaje electrónico impone la necesidad de buscar alternativas a los métodos tradicionales para la identificación de estilos de aprendizaje. Dos premisas deseables, en una alternativa potencial, son la capacidad de realizar la identificación en tiempo real, y el hacerlo sin información previa, es decir, sin emplear cuestionarios, como es en el caso de EDUCA.

6. Zamná: una red social de aprendizaje adaptativo

Las redes sociales continúan en gran auge en nuestros días, y a medida que van creciendo sus funcionalidades, es mayor el número de personas que se unen a ellas. Las redes sociales ayudan a las personas a mantenerse en contacto con otras, compartir información o socializar, entre otras cosas, y su ámbito de uso o aplicaciones se ha ido expandiendo desde los más comunes, como el ámbito familiar o social, hasta áreas académicas o de negocios.

Esta herramienta tan efectiva para que personas, grupos y comunidades realicen intercambios dinámicos, puede ser utilizada también a favor de la educación, y así crear un ambiente de aprendizaje colaborativo donde los usuarios puedan aportar sus conocimientos académicos a toda la red.

Por otra parte, y en relación particular con la educación, reconocer y clasificar las características del aprendizaje de los estudiantes es un trabajo complejo. Como ya se mencionó con anterioridad, un enfoque interesante para identificar y clasificar los estilos de aprendizaje es la teoría de Felder & Silverman (1998) que clasifica los estilos de aprendizaje en cuatro dimensiones que son: global-secuencial, visual-verbal, sensitivo-intuitivo y activo-reflexivo.

6.1 La web 2.0

Las nuevas tecnologías de aprendizaje como: la educación electrónica, móvil y colaborativa (*e-, m-, c-Learning*, respectivamente) no se aprovechan adecuadamente en la gran mayoría de escuelas en México. Muchos de los sistemas actuales que permiten diseñar y crear materiales educativos están basados en un autor que los crea y muchos usuarios que acceden a ellos de manera personal, sin reconocer la individualidad que manifiesta cada sujeto para aprender. En las aplicaciones llamadas web 2.0, son los usuarios quienes crean los contenidos que se ven en dichos sitios y cada día se incrementa el número de usuarios interesados en participar.

La web 2.0 fue una evolución de la web que inició con una sesión de lluvia de ideas entre O'Reilly y MediaLive International (O'Reilly, 2006). La web 2.0 se enfoca más en tomar en cuenta al usuario final, de modo que estos generan trabajos en forma colaborativa.

La contribución del usuario da origen al término de “inteligencia colectiva”. Esta viene a ser uno de los aspectos principales de la revolución web, toda vez que se confía en gran parte en lo que los usuarios aportan y se asume que el sitio mejorará entre más gente lo utilice o añada nuevo contenido. El cambio de la web sencilla (o 1.0) a lo que se conoce como web 2.0 no fue un proceso fácil, pero permitió que muchas empresas innovaran sus modelos de negocio respecto a la forma en que trabajaban antes. Otra característica de la web 2.0 es que no está limitada a un solo dispositivo ni una plataforma específica, aunque la web 1.0 ya no se limitaba a un único dispositivo. La web 2.0 toma un agregado a esta definición, ya que con el ingreso de las arquitecturas orientadas a servicios se expande el concepto. En la tabla 3 se listan unos ejemplos de transición de la web 1.0 a la web 2.0 según O'Reilly (2006).

Tabla 3. Comparación de aplicaciones web 1.0 y su similar en la web 2.0

<i>Web 1.0</i>	<i>Web 2.0</i>
Doble click	Google AdSense
Ofoto	Flickr
Akamai	BitTorrent
mp3.com	Napster
Britannica Online	Wikipedia
Sitios personales	Blogging
Evite	upcoming.org and EVDB
Especulación en el nombre de dominio	Optimización del motor de búsqueda
Páginas visitadas	Costo por clic
Pantalla raspada	Servicios web
Publicaciones	Participación
Sistemas manejadores de contenidos	Wikis
Directorios (taxonomía)	Etiquetado ('folksonomía')
Adhesiones	Sindicación

Fuente: O'Reilly (2006).

La lista anterior solo es una pequeña muestra de la inmensidad de sitios que nacen día a día, aunque cabe mencionar que sitios como Amazon, eBay, YouTube no han sido mencionados pero son parte del gran avance que tiene la web hoy en día.

6.2 ¿Qué es Zamná?

Zamná es una red social de aprendizaje adaptativo, aunque también se podría ver como una herramienta de autor para generar sistemas tutores inteligentes. Con ella se pretende que existan diferentes tipos de usuarios en la red, como se describe a continuación:

- **Usuario administrador de la red social en el servidor**, encargados de efectuar los procesos de gestión y administración del sistema.
- **Usuarios padres/madres de estudiantes**, quienes entrarían a la red para supervisar o simplemente ver el desempeño de sus hijos.
- **Usuarios profesores**, los cuales tienen como misión crear los cursos o material de aprendizaje.
- **Usuarios estudiantes**, los actores más importantes en el sistema pues son quienes acceden a la red para socializar y también para aprender.

Zamná (Zatarain-Cabada *et al.*, 2010) funciona bajo un ambiente de operación web 2.0. Esta plataforma de aprendizaje reúne las ventajas de ser adaptativa e inteligente, además de operar dentro de un contexto de inteligencia colectiva que establece un esquema donde los usuarios son los generadores de su propio conocimiento.

La principal característica de Zamná es que aprovecha las nuevas tecnologías de información y comunicación desde una perspectiva híbrida, de tal manera que se pueden combinar los métodos tradicionales de enseñanza en la escuela con las nuevas tecnologías de aprendizaje, incluyendo el uso de las redes sociales de internet y otros elementos que constituyen la web 2.0. Para aplicar estas tecnologías en el ámbito educativo, la herramienta Zamná cuenta con un ambiente de aprendizaje inteligente a través de una red social que permite reunir a los agentes del proceso educativo bajo un ambiente de cooperación y colaboración, con el fin de propiciar el desarrollo personal y social de los participantes, mejorando así el nivel educativo de los usuarios.

Los usuarios de la red social de aprendizaje, siguiendo el enfoque colaborativo, podrán crear y actualizar cursos o materiales educativos de diversos temas, los cuales estarán disponibles para todos los miembros de la comunidad. Estos materiales educativos tendrán características especiales que durante su ejecución permitirán, de manera inteligente y dinámica, adaptar su contenido de acuerdo con el estilo de aprendizaje propio del estudiante, el cual será previamente identificado y almacenado en el perfil del usuario.

Las principales contribuciones de Zamná son: creación de una red social de aprendizaje, para que una comunidad de usuarios pueda producir y utilizar cursos adaptativos e inteligentes; propiciar la interacción entre usuarios que a su vez permita aprovechar el potencial del aprendizaje colaborativo; ofrecer diferentes tipos de interfaces gráficas que presentan una gama especial de servicios orientados al usuario; proveer un enfoque educativo híbrido donde se puedan combinar los métodos tradicionales de enseñanza en la escuela con las nuevas tecnologías de enseñanza y aprendizaje, incluyendo entre estas el uso de las redes sociales y otros elementos que constituyen la web 2.0.

6.3. Análisis y diseño de Zamná

El proceso de desarrollo del proyecto está basado en el ciclo de desarrollo de un sistema de *software*. Los procesos de desarrollo de un sistema conllevan por lo general las etapas de análisis y requerimientos, diseño, implementación, validación o pruebas y mantenimiento. En esta sección se describirán las etapas de análisis y diseño del *software* de la herramienta.

Análisis de requerimientos

El objetivo de esta etapa de desarrollo es establecer los requerimientos tanto funcionales como de calidad que debe cumplir el sistema. Para ello se estableció un conjunto de requerimientos de usuario, los cuales fueron analizados para determinar su prioridad e importancia en el sistema, así como su clasificación en cuanto a funcional o calidad.

Los principales casos de uso se obtuvieron considerando los requisitos funcionales. Estos casos de uso se derivaron de un extenso estudio realizado a diversas de redes sociales populares como Facebook y MySpace, así como redes sociales de aprendizaje como LearnHub y LectureShare. En este estudio se investigó cuáles eran las principales funcionalidades que operan en dichas redes sociales. Además, se realizó otro estudio sobre las familias de los modelos de estilos de aprendizaje. Con el resultado de ambos estudios se establecieron los requerimientos de calidad (véase tabla 4) y funcionales requeridos para una red social de aprendizaje inteligente. Los requerimientos funcionales se utilizaron para definir los casos de uso del sistema, y se presentan en la tabla 5.

Tabla 4. *Requisitos de calidad de Zamná*

<i>Id</i>	<i>Nombre</i>	<i>Descripción</i>
RC1	Usabilidad	El sistema debe presentar al usuario una interfaz gráfica amigable y fácil de usar. La navegabilidad en el sitio web debe ser intuitiva para el usuario.
RC2	Seguridad	El sistema debe garantizar la seguridad y confidencialidad de los datos del usuario.
RC3	Mantenimiento	La tarea de mantenimiento del sistema debe realizarse con bajo esfuerzo. Si se desea agregar nuevos componentes al sistema deben realizarse cambios mínimos en la implementación.

Tabla 5. *Casos de Uso de Zamná*

<i>Id</i>	<i>Función</i>	<i>Descripción</i>
UC1	Registrar Usuario	El usuario se registra en el sistema por primera vez ingresando su información personal.
UC2	Crear Perfil de Usuario	El usuario crea su perfil de usuario introduciendo su información personal y gustos.
UC3	Editar Perfil de Usuario	El usuario modifica la información actual contenida en su perfil de usuario.
UC4	Cambiar Contraseña	El usuario modifica la contraseña establecida en su registro de usuario.
UC5	Contestar Test de Estilo de Aprendizaje	El usuario contesta el cuestionario de estilos de aprendizaje de Felder-Silverman para obtener su perfil de estilo de aprendizaje.
UC6	Crear Curso	El usuario crea un curso en el editor de cursos con contenidos de aprendizaje.
UC7	Visualizar Curso	El usuario visualiza el contenido de un curso seleccionado.
UC8	Calificar Curso	El usuario califica el contenido de un curso.
UC9	Editar Curso	El usuario modifica el contenido de un curso.
UC10	Cargar Curso	El usuario sube a la red el archivo correspondiente a un curso creado.
UC11	Publicar Curso	El usuario publica un curso que ha sido cargado.
UC12	Crear Comunidad	El usuario crea una comunidad.
UC13	Unirse a Comunidad	El usuario se une a una comunidad y pasa a ser miembro de la misma.
UC14	Crear un Tema de Discusión	El usuario crea un tema de discusión introduciendo el título y su comentario.
UC15	Comentar un Tema de Discusión	El usuario introduce un comentario en un tema de discusión ya creado.
UC16	Cargar Recursos	El usuario sube a la red archivos tipo documento.
UC17	Publicar Recursos	El usuario publica un documento que ha sido cargado.
UC18	Crear Noticia	El usuario crea una lección.
UC19	Publicar Noticia	El usuario publica una lección creada.

Diseño

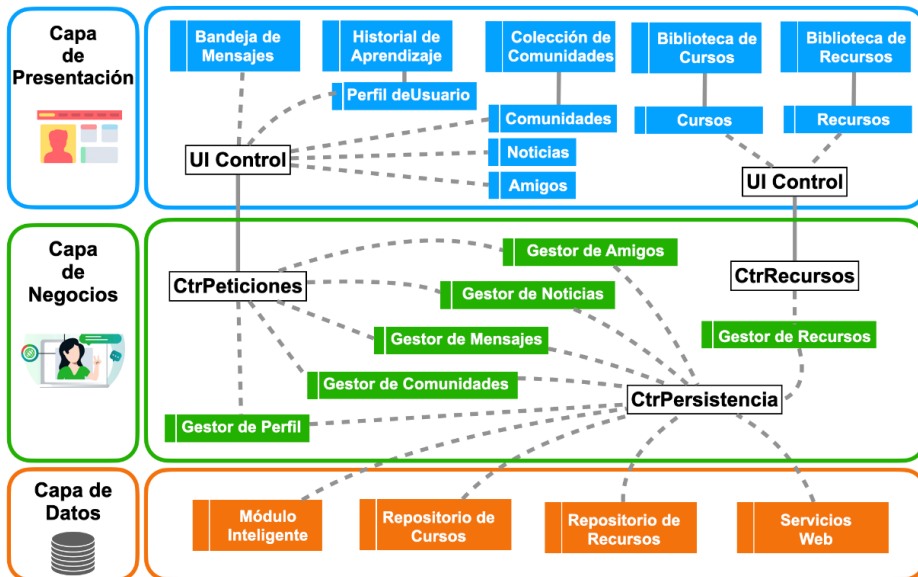
El diseño del sistema se realizó tomando en cuenta los requisitos recabados en la etapa de análisis, con base en los casos de uso identificados. La etapa de diseño se realizó usando la metodología QASAR (Bosch, 2000), gracias a la cual se estableció la arquitectura del sistema, refinando los componentes en iteraciones sucesivas. Los componentes principales de la herramienta son: el editor de cursos, el visor de cursos en web y el módulo inteligente,

además de la interfaz gráfica de usuario (sitio web) y la base de datos. Después del diseño, se procedió a implementar estos componentes utilizando tecnologías y lenguajes de programación tales como Flex, Apache, PHP y MySQL.

6.4 Arquitectura de la red social educativa

El diseño arquitectónico para la aplicación basado en la web se centra en la definición de una estructura global, representada en una arquitectura por capas. El modelo de capas organiza el sistema en tres componentes principales: **Capa de Presentación**, **Capa de Negocios** y **Capa de Datos**. Cada capa proporciona un conjunto de servicios a las otras capas o al usuario. La figura 30 muestra la arquitectura en tres capas diseñada para el sistema.

Figura 30. Arquitectura en capas de la plataforma Zamná



A continuación, se describe el funcionamiento de cada capa de la arquitectura del sistema.

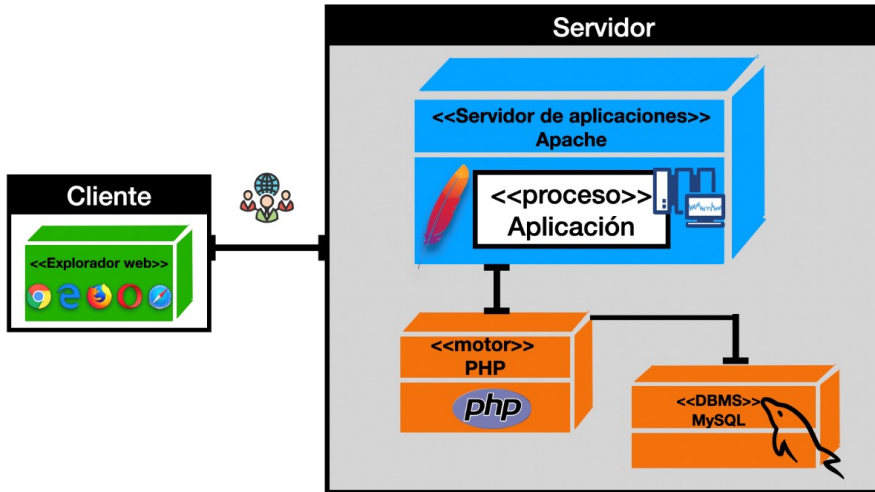
Capa de datos. En esta capa se encuentran alojados los componentes encargados del control de la información. El Módulo Inteligente contiene, envía y recibe datos que determinan el estilo de aprendizaje del estudiante que usa el sistema. Existen dos repositorios (Repositorio de Cursos y Repositorio de Recursos) que almacenan la información de los cursos y los recursos que se utilizan en los cursos, además del componente de Servicios Web. Todos estos componentes se comunican con la capa de negocios a través del componente que controla la persistencia de los datos.

Capa de negocios. Esta capa contiene los controladores de comunicación entre los elementos de la capa de presentación y los de la capa de datos. Existen tres componentes controladores (CtrPeticones, CtrPersistencia y CtrRecursos), los cuales coordinan las peticiones de gestión de datos que realizan los seis componentes gestores necesarios para administrar la información de Perfil, Comunidades, Mensajes, Noticias, Amigos y Recursos. Los componentes CtrPeticones y CtrRecursos se encargan de establecer la comunicación entre la capa de servicios y la capa de presentación.

Capa de presentación. La capa contiene dos controladores de interfaz, llamados UIControl, que se encargan de dirigir las operaciones de trabajo al componente de interfaz que corresponde. UIControl (izquierda) se comunica con los componentes relacionados con la información que el usuario utiliza en la interacción (Bandeja de Mensajes, Perfil de Usuario, Comunidades, Noticias y Amigos), mientras que el componente Perfil de Usuario se comunica a su vez con el Historial de Aprendizaje, y el de Comunidades con el componente Colección de Comunidades. El segundo UIControl (derecha) controla las operaciones de gestión de recursos de la red, como los cursos y los recursos a través de los componentes Cursos, Recursos, Biblioteca de Cursos y Biblioteca de Recursos.

La figura 31 presenta el diagrama de despliegue que representa los diferentes elementos que conforman la arquitectura física sobre la que se ejecuta la aplicación usando el modelo cliente-servidor. A continuación se describen esos componentes.

Figura 31. Diagrama de despliegue



Cliente: Representa el dispositivo físico (computadora personal) que utiliza el usuario para acceder a la aplicación a través de un explorador web.

Servidor: Representa el equipo de cómputo que recibe las peticiones del cliente y aloja los servicios necesarios para procesar dichas peticiones. Está compuesto de tres elementos: Apache, Motor PHP y MySQL.

Interfaz gráfica

La interfaz gráfica de usuario es parte fundamental de la red social. En la figura 30 se puede observar que la capa de presentación está compuesta por siete componentes principales: Bandeja de Mensajes, Perfil de Usuario, Comunidades, Noticias, Amigos, Recursos y Cursos. El componente Cursos es muy importante en la red social porque es el que se encarga de administrar los cursos creados en esta red. Este componente, a su vez, incorpora dos componentes llamados Editor de Cursos y Visor de Cursos, los cuales se utilizan para crear y visualizar los cursos que se comparten en la red social. El Editor de Cursos es en donde se crean los tutores inteligentes que después serán desplegados por el Visor de Cursos adaptándose al estilo de aprendizaje del usuario.

Para el diseño e implementación de la interfaz gráfica de usuario se utilizó la tecnología Flex de Adobe. La selección de Flex se basó en que permite a los desarrolladores de aplicaciones web construir rápida y fácilmente aplicaciones ricas de internet (RIA por sus siglas en inglés). En una arquitectura multicapa como la que se usó para el diseño de los componentes de Zamná, la aplicación Flex se utiliza en el nivel de presentación. Otra ventaja considerada es que Flex es un marco de trabajo gratuito y altamente productivo para la creación y mantenimiento de aplicaciones web, el cual se implanta en los principales exploradores, como Internet Explorer, Mozilla Firefox, Google Chrome y los distintos sistemas operativos.

Base de datos

Una parte fundamental del sistema es la administración adecuada de los datos de usuario. El manejo de la persistencia de la información, generada por cada uno de los usuarios de Zamná, es de suma importancia para el buen funcionamiento del sistema. Los usuarios registran a través de la interfaz gráfica de la web sus datos de usuario y perfil, además de que suben archivos tales como documentos y cursos que deben ser almacenados para que se encuentren disponibles en la red y así puedan ser consultados.

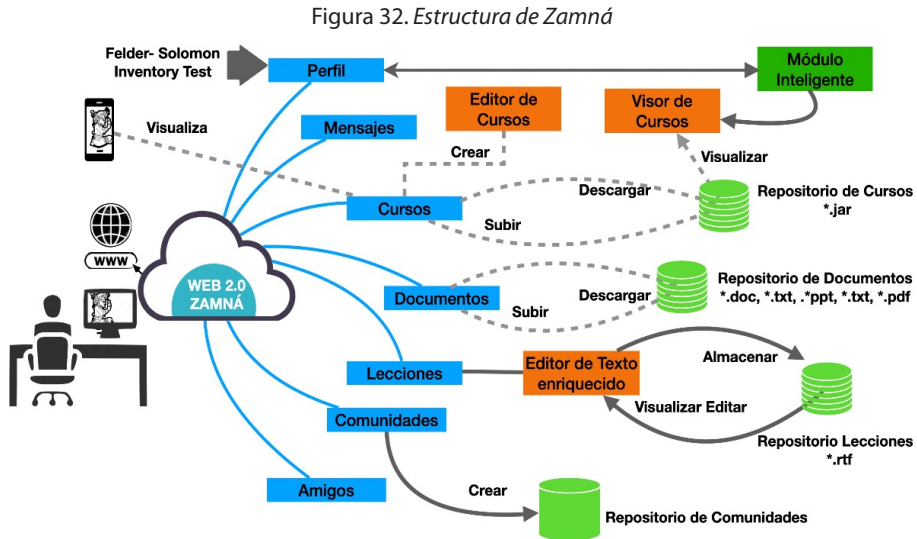
La base de datos de Zamná utiliza el sistema manejador de base datos MySQL debido a que es reconocido como el DBMS de código abierto más popular, confiable y fácil de usar, y corre en más de 20 plataformas, incluidas Windows y Linux. Además, es usado para desarrollo en la web, combinando Apache como servidor web y PHP como lenguaje de programación.

6.5 Navegación en el sistema Zamná

La figura 32 ilustra los elementos del sistema con los que los usuarios pueden interactuar durante una sesión en Zamná. Un usuario entra en el sitio (Zamná) usando cualquier navegador web desde su dispositivo. A continuación, el usuario puede acceder a su perfil, crear cursos, comunidades y lecciones, subir documentos, hacer contactos y amigos, etc.

Los cursos creados por el usuario son adaptativos (sistemas tutores in-

teligentes) para los estudiantes, quienes aprenden bajo diferentes estilos de aprendizaje de acuerdo con el modelo de Felder-Silverman. En la misma figura 32 se observa que un curso creado con Zamná se puede exportar a un dispositivo móvil o puede ser visualizado desde un navegador web.



Comunidades o sociedades del conocimiento

Una de las principales virtudes de Zamná es la funcionalidad de comunidades. Con esta se puede crear una especie de sociedad de conocimiento donde se agrupan diferentes miembros de la red social.

Una sociedad o comunidad de conocimiento contiene, entre otros elementos, sistemas tutores inteligentes, clases y materiales educativos utilizados por esta comunidad. Por ejemplo, una sociedad de conocimiento de Compiladores contiene todos los materiales de aprendizaje que los usuarios de esta sociedad suben, crean, comparten y estudian.

La figura 33 muestra algunas interfaces del ambiente de Zamná para los procesos de:

1. Crear una comunidad (arriba izquierda).
2. Crear curso (arriba derecha).
3. Navegar por un curso (abajo izquierda).
4. Responder un examen (abajo derecha).

Figura 33. Interfaces de Zamná



Ejemplo de comunidad

Un profesor del área de Programación de Sistemas puede crear una comunidad que congrege a otros profesores del área, además de estudiantes inscritos en algún curso relacionado o interesados en dicha área de conocimiento.

Dentro de la comunidad, todos los miembros pueden subir documentos o archivos, recomendar y compartir cursos y material interesante para la comunidad, además de crear foros de discusión sobre problemas o asuntos relacionados con los temas de estudio. La figura 34 presenta la interfaz principal de una comunidad llamada Programación de Sistemas.

Figura 34. Comunidad de Programación de Sistemas

The screenshot shows the main interface of the Zamná community for 'Programación de Sistemas'. At the top, there is a header with the community name, a user profile for Adan Jose (adanjoga), and a 'Cerrar sesión' button. Below the header is a navigation menu with options: Inicio, Cursos, Noticias, Recursos, Miembros, and Foro. A central banner area contains a mobile phone icon, a description of the community's purpose, and the 'Autor de Comunidad: Ramon Zatarain Cabada, Doctor, rzc777@hotmail.com'. The main content area is divided into several sections:

- + Cursos recientes:** A list of uploaded courses including 'Compiladores' (2010-06-06), 'Crear una comunidad' (2010-06-06), 'CURSO ProgSist.ppt' (2010-02-04), 'CAP2,3.doc' (2010-02-04), 'javacc-5.0.zip' (2010-02-17), 'BNF for MicroJava1.doc' (2010-03-24), 'Javita.jj' (2010-04-13), 'ExámenesPS.rar' (2010-05-21), 'Demo1.java' (2010-06-01), and 'Demo1.txt' (2010-06-01).
- + Noticias recientes:** A list of recent news items.
- + Recursos recientes:** A list of recent resources.
- + Miembros de Programación de Sistemas:** A grid of member avatars.
- + Discusiones & Debates:** A list of discussion topics including 'Compilador de compiladores Javacc' (2010-02-17), 'Proyecto # 1 Scanner y Parser de gramatica micri' (2010-03-24), 'JAVACC' (2010-04-08), 'TIPS para todos' (2010-04-17), and 'Java Bytecode' (2010-06-01).
- + Recomendaciones de Programación de Sistemas:** A list of recommended content.

En la parte de arriba a la izquierda de la figura se pueden apreciar los cursos subidos, en la parte de arriba (debajo del título de la comunidad) está un menú de opciones para visualizar elementos de la comunidad como Cur-

sos, Noticias, Recursos, Miembros y Foro. En la parte central de la imagen a la izquierda, aparecen los cursos recientes que se han incorporado, y en la parte derecha, aparece un listado de recursos agregados, los cuales pueden tener diferentes formatos (ppt, doc, txt, zip, java, etc.). En la sección de abajo a la izquierda se observan los miembros de la comunidad y a un lado las discusiones o foros de debate que mantiene esa comunidad.

Evaluación adaptativa de aprendizajes

Respecto a la evaluación de una sociedad o comunidad, el objetivo es aumentar sus conocimientos en un área particular de estudio y colocar a sus integrantes en un nivel nuevo y superior. El método actual para obtener información de una comunidad es mediante la publicación de las pruebas individuales con características de adaptación para cada estudiante. Este instrumento es una prueba objetiva y formativa para retroalimentar a la persona y evaluar los resultados del grupo.

En primer lugar, se utilizó el estilo de aprendizaje del usuario identificado en Zamná para definir el proceso de adaptación. Se usaron también las técnicas y métodos de adaptación para la construcción de exámenes adaptativos por elementos de agrupación en función del material de aprendizaje asociado.

Para definir el instrumento adaptativo para evaluación, se desarrollaron varias versiones de una sola pregunta que presenta el material multimedia en forma alternativa (es decir, audio, video o texto) de acuerdo con el estilo de aprendizaje de los usuarios identificados previamente en Zamná. También es posible clasificar los elementos según el nivel de complejidad para llevar a cabo un segundo proceso de adaptación. Una vez que el estudiante realiza una actividad de aprendizaje en Zamná, también realiza una prueba formativa para medir los conocimientos adquiridos y contribuir a los resultados del grupo.

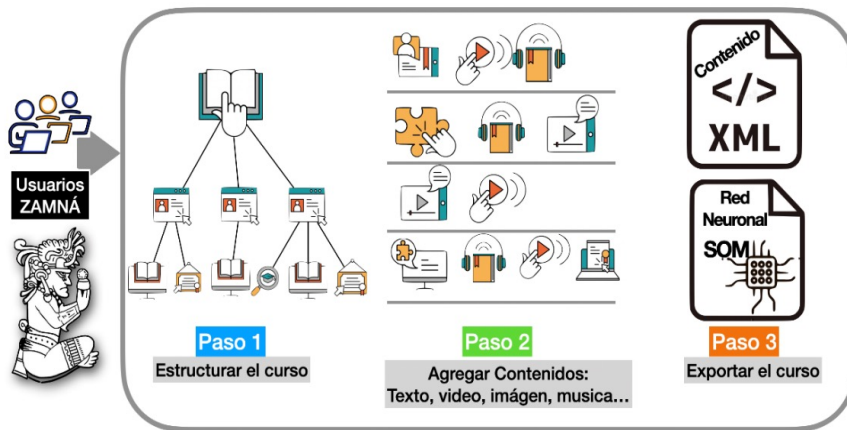
Editor de cursos

El editor de cursos permite al usuario crear cursos que sean publicados en la red. El editor de cursos se diseñó para que el usuario pueda agregar dife-

rentes tipos de elementos al curso, como imágenes, video o texto, a los cuales se les asigna un valor que corresponde a alguno de los diferentes estilos de aprendizaje establecidos por Felder-Silverman.

La figura 35 ilustra el proceso instrumentado para la creación de un curso.

Figura 35. Proceso de creación de un curso



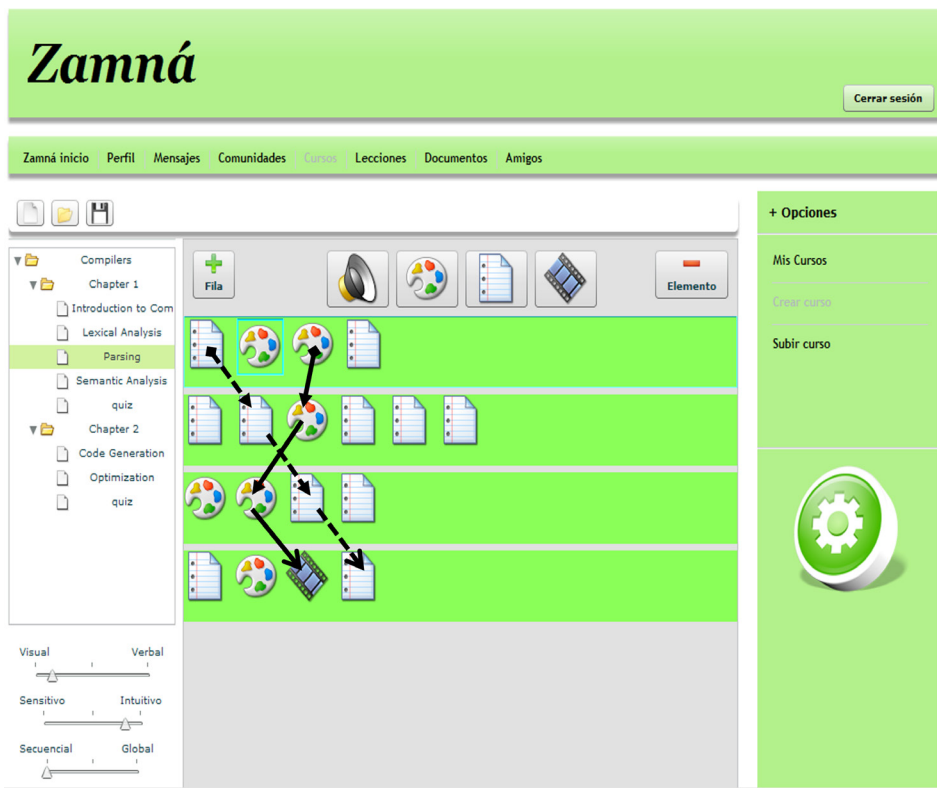
Los usuarios son los actores principales en el proceso de creación de un curso en Zamná. El curso puede ser un curso adaptativo o sistema tutor inteligente (STI) que se genera a través de un editor visual incluido en la herramienta. El primer paso para crear un curso es generar su estructura de acuerdo con una jerarquía de temas y subtemas. El segundo paso consiste en agregar elementos (componentes preparados en formatos multimedia como archivos de audio, video, imágenes o textos) que pueden ser importados desde archivos externos, o bien el autor puede escribir material didáctico usando directamente el editor visual.

Cuando el autor agrega los materiales de aprendizaje al curso, asigna a cada material un valor de acuerdo con el estilo (tipo) de aprendizaje (visual-verbal, deductivo-intuitivo y secuencial-global) del modelo de Felder y Silverman. Una vez agregados los materiales de cada tema, el tercer paso del proceso es generar un archivo XML que corresponda al curso estructurado. Este archivo permite exportar el curso a la web, además de ser exportado a un dispositivo móvil, ya que dentro del archivo contenedor se inclu-

ye el intérprete XML del curso. Este archivo contiene también el módulo inteligente que implementa un mapa autoorganizado (SOM por sus siglas en inglés), el cual tiene la tarea de desplegar el material del curso en la web o el dispositivo móvil, según el estilo de aprendizaje del usuario.

La figura 36 muestra la interfaz durante la edición de un curso de Compiladores. En el ejemplo se aprecian dos diferentes trayectorias para la presentación de temas a los estudiantes.

Figura 36. Interfaz del editor de cursos



Dentro de la figura 36, en la parte izquierda aparece la estructura (árbol) del curso, donde el nodo raíz llamado “Compilers” tiene dos nodos hijos llamados “Chapter 1” y “Chapter 2”, los cuales a su vez contienen los temas y el examen (*quiz*) que forman parte del capítulo. Debajo de esta

estructura aparecen tres controles para que manualmente el autor del curso defina el estilo de aprendizaje de cada recurso de aprendizaje editado. En la parte central de la interfaz aparecen los diferentes recursos de aprendizaje (sonido, imagen, texto, video), los cuales permiten agregar elementos a cada tema. Cada línea representa un tema y contiene varios recursos (la primera línea contiene cuatro) los cuales se crearon de acuerdo con un estilo de aprendizaje definido por el autor. Para ejemplificar lo referente a la parte inteligente o adaptativa de los cursos, en la figura aparecen dos flechas distintas que indican los diferentes caminos que podría seguir la presentación de los temas, de acuerdo con el estilo de aprendizaje del estudiante. Por ejemplo, un estudiante visualizaría el material (un recurso de cada tema) señalado con la flecha punteada, el cual presentaría los recursos 1, 2, 3 y 4; mientras otro estudiante visualizaría o estudiaría los recursos 3, 3, 2 y 3 que corresponden a la flecha continua.

Módulo inteligente

Se implementó un módulo inteligente para identificar el estilo de aprendizaje de los estudiantes. Este módulo inteligente contiene una red neuronal entrenada para identificar los estilos de aprendizaje de los estudiantes que visualizan los cursos en web o dispositivos móviles. El entrenamiento de la red neuronal se realizó utilizando los datos recabados de una serie de experimentos. Dichos experimentos se llevaron a cabo con estudiantes de nivel bachillerato. Los datos recolectados se dividieron en dos partes: datos de entrenamiento y datos de prueba (conformados por 85 y 15% de los datos totales respectivamente).

La red neuronal implementada es un mapa auto-organizado (SOM) o mapa de Kohonen. Cuando la red neuronal recibe un nuevo patrón de entrada, cada una de las neuronas obtiene una copia de este a través de las conexiones entre las neuronas que se encuentran en la capa de entrada y las que están ubicadas en la capa de Kohonen. Las conexiones laterales sirven para proveer una retroalimentación a las neuronas vecinas, la cual puede ser de excitación o inhibición, dependiendo de la distancia a la que se encuentre de la neurona ganadora.

El módulo inteligente se integra con varios componentes para realizar su trabajo. Por ejemplo, el módulo de perfil del estudiante hace uso del módulo inteligente para obtener el estilo de aprendizaje del usuario; mientras que el visor de cursos es responsable de mostrar los contenidos de un curso de acuerdo con el estilo de aprendizaje definido por el módulo inteligente (ver figura 35).

7. Fermat: sistema para la enseñanza de matemáticas

La computación afectiva es uno de los tópicos más influyentes en los diferentes campos del aprendizaje y representa una de las áreas de interés más activas en las conferencias sobre tecnologías del aprendizaje, como en los siguientes eventos internacionales: Intelligent Tutoring Systems Conference (ITS), Artificial Intelligence in Education (AIED), e International Conference in Advanced Learning Technologies (ICALT). El tema de tecnologías inteligentes y afectivas para el aprendizaje involucra a investigadores de diversos campos como computación, inteligencia artificial, psicología y pedagogía. En las últimas dos décadas, un número creciente de sistemas tutores inteligentes y afectivos han sido desarrollados para apoyar la enseñanza en diferentes campos de estudio (Hasan *et al.*, 2020; Petrovica *et al.*, 2017).

La investigación en el área de computación afectiva incluye tanto la detección de emociones de los usuarios como la respuesta que los sistemas emiten a partir de ellas. Los sistemas de detección de emociones son capaces de captar diversas señales que emite el usuario a través del rostro, el habla, la conversación escrita y otros rasgos humanos que comunican estados emocionales como frustración, interés, aburrimiento, entre otros. Estos sistemas reciben señales de sensores como cámaras web (Essa & Pentland, 1995; Yacoub & Davis, 1996), micrófonos (Tosa & Nakatsu, 1996) y texto a través de diálogos conversacionales (Graesser *et al.*, 2004), entre otros. Por otra parte, los sistemas de reconocimiento de emociones implementan for-

mas eficientes para gestionar las emociones negativas del estudiante (Arroyo *et al.*, 2009; D’Mello *et al.*, 2007; Boulay, 2011).

En este capítulo se presenta un sistema al que se integraron redes sociales de aprendizaje, sistemas tutores inteligentes y computación afectiva para la enseñanza de las matemáticas. Un sistema tutor inteligente fue incorporado a una red social de aprendizaje llamada Fermat. Este sistema se desarrolló con el objetivo principal de apoyar a los estudiantes mexicanos a mejorar sus resultados en la Evaluación Nacional del Logro Académico en Escuelas de México (ENLACE¹). La prueba ENLACE es una evaluación estándar del sistema de educación nacional en la que participan estudiantes de educación básica (primaria y secundaria). En 2011, 14 millones de niños de primaria y secundaria participaron en la prueba y los resultados revelaron que más de nueve millones de ellos tienen un nivel insuficiente en aprendizaje en matemáticas.

7.1 La enseñanza de matemáticas en México

En México, un estudiante de educación básica (primaria) asiste durante 40 semanas a clases, acumulando un aproximado de 800 horas de clases al año. El número de estudiantes por grupo ha disminuido en los últimos años; actualmente cada maestro de nivel primaria atiende en promedio a 25 estudiantes (SEP, 2021). En la figura 37 se presenta un ejemplo del número de horas lectivas en el ciclo escolar 2011-2012 para los estudiantes de tercer año de primaria.

De esta cantidad de horas en el ciclo escolar 2011-2012, un profesor impartía un promedio de 7.2 diferentes áreas o materias a los estudiantes. Los maestros deben seguir el programa de estudio oficial que contiene los temas de estudio y algunas recomendaciones de cómo enseñar los contenidos. A continuación, se describen dos pruebas que se aplican a nivel nacional e internacional para medir el nivel de conocimiento que poseen los estudiantes de educación básica.

¹ Información en <http://www.enlace.sep.gob.mx/>

Figura 37. Horas lectivas de estudiantes de tercer año de primaria

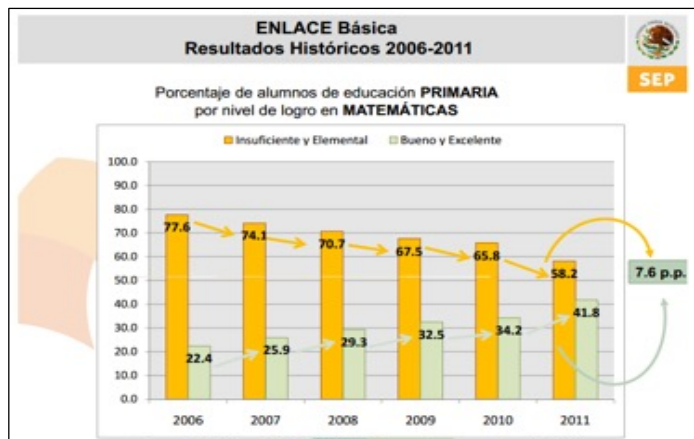
Tercer grado de educación primaria		Periodos lectivos por semana
Asignaturas ACUERDO 592	Español	5
	Segunda Lengua: Inglés	2.5
	Matemáticas	5
	Ciencias Naturales	2
	La Entidad donde Vivo	2
	Formación Cívica y Ética	1
Áreas de Desarrollo personal y social ACUERDO 12/10/17	Educación Física	1
	Artes	1
	Educación socioemocional	0.5
TOTAL		20
Autonomía curricular ACUERDO 12/10/17	Periodos lectivos mínimos	2.5

Fuente: DOF (2011).

Prueba ENLACE

La Evaluación Nacional de Logro Académico en Centros Escolares (ENLACE) se aplicaba anualmente tanto en planteles públicos como privados, con el fin de medir el nivel de dominio en las áreas de español, matemáticas y una tercera materia académica que difiere cada año. En la figura 38 se muestran los resultados de la prueba aplicada desde 2006 hasta 2011.

Figura 38. Porcentaje de resultados 2006-2011



Fuente: ENLACE (2011).

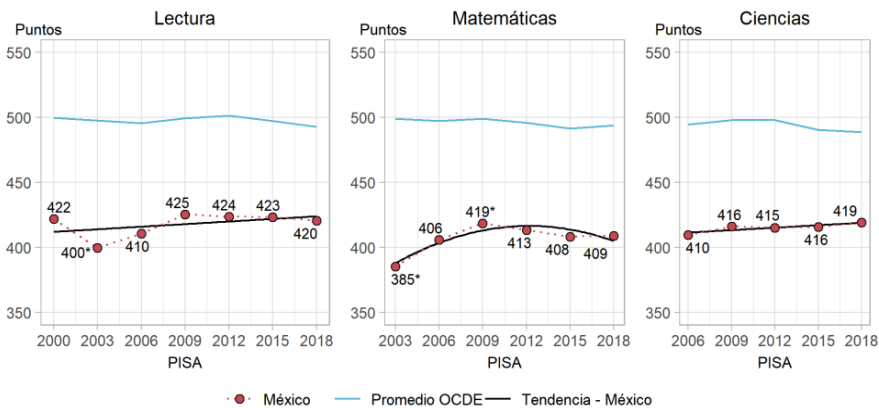
Aunque los resultados de la prueba ENLACE, como se observa en la figura 38, mejoraron con los años, los estudiantes catalogados con conocimiento Insuficiente y Elemental son la mayoría (58.2% en 2011), mientras que solo 41.8% está en los niveles Bueno y Excelente. Se aprecia un bajo rendimiento de los alumnos de primaria en matemáticas, lo que subraya que se realicen cambios en las estrategias de enseñanza para poner un mayor énfasis en el individualismo del aprendizaje.

Prueba PISA

El Programa Internacional de Evaluación de Estudiantes (PISA, por sus siglas en inglés) es un estudio periódico y comparativo que promueve y organiza la Organización para la Cooperación y el Desarrollo Económico (OECD por sus siglas en inglés), y en ella participan los países miembros y no miembros (asociados) de la organización. Esta prueba se aplica cada tres años desde 1997 y en México desde el año 2000.

En esta evaluación participan estudiantes de 15 años de más de 60 países en el mundo. PISA tiene la tarea de evaluar tres áreas: matemáticas, ciencias y lectura. Con esta prueba se busca conocer en qué medida los estudiantes de 15 años han adquirido los conocimientos y habilidades necesarios para participar activa y plenamente en la sociedad moderna. La figura 39 presenta la tendencia en el desempeño de los estudiantes mexicanos de 2000 a 2018.

Figura 39. Tendencias en el desempeño en lectura, matemáticas y ciencias



Nota: *Indica que el rendimiento promedio estimado está, en términos estadísticos, significativamente por encima o por debajo de las estimaciones de PISA 2018.
 La línea azul indica el rendimiento medio promedio en todos los países de la OCDE con datos válidos en todas las evaluaciones PISA. La línea punteada roja indica el rendimiento medio en México. La línea negra representa una línea de tendencia para México (línea de mejor ajuste).
 Fuente: OECD, PISA 2018 Database, Tables I. B1.10, I. B1.11 and I. B1.12.

Como se puede apreciar en estos resultados, la tendencia en todas las áreas de evaluación no presenta una mejora significativa a través de los años, y los resultados de los estudiantes mexicanos están muy por debajo de la media de los países de la OCDE. Esto indica que los estudiantes mexicanos están en desventaja para resolver situaciones que se les presenten en la vida real. La educación básica debe proporcionar bases sólidas para que los estudiantes sean capaces de enfrentar situaciones de este tipo y demuestren competencias para desempeñarse favorablemente en la vida real.

Los resultados que se muestran en las pruebas de ENLACE y PISA son un indicio de la deficiente formación de los alumnos en un área primordial como las matemáticas.

7.2 ¿Qué es Fermat?

Fermat es una aplicación web que se compone de dos elementos principales: una red social de aprendizaje y un sistema tutor inteligente (STI). La red social de aprendizaje es un espacio social en la web donde los diferentes miembros de la red (profesores, estudiantes y padres de familia) comparten los recursos y funcionalidades que una red social otorga. Por otra parte, el STI permite al estudiante aprender matemáticas de una manera personalizada, como lo lleva a cabo un tutor humano, aplicando técnicas vanguardistas de computación afectiva para considerar no solo aspectos cognitivos del estudiante, sino también estados emocionales o afectivos.

Para llevar a cabo la evaluación de los elementos cognitivos del alumno, se implementó un sistema experto, el cual facilita la elección del siguiente ejercicio del estudiante, considerando el aprendizaje o conocimiento que haya logrado. Respecto a los elementos afectivos, Fermat realiza el reconocimiento de emociones del estudiante captando señales a través de una cámara (imagen del rostro) y un micrófono web (audio de voz). El estado emocional captado es importante para tomar decisiones cuando se selecciona un ejercicio que debe ser resuelto por el estudiante. Para esto, Fermat se apoya en un agente pedagógico, el cual interactúa con el alumno.

7.3 Análisis, diseño e implementación de Fermat

Al igual que con los dos proyectos de *software* presentados en los capítulos 5 y 6 (EDUCA y Zamná), para el desarrollo de la herramienta Fermat se aplicó principalmente una metodología orientada a objetos. Antes de aplicar esta metodología, se realizó una investigación de los métodos más comunes y populares en la enseñanza de los temas de multiplicación y división de números enteros, correspondientes al tercer año de primaria. Este análisis generó como producto la especificación de requerimientos para el desarrollo de dos sistemas tutores inteligentes inmersos dentro de Fermat, los cuales se encargan de enseñar estos temas (multiplicación y división de números enteros). La etapa de análisis es de vital importancia para el desarrollo de cualquier aplicación de *software*, puesto que es donde se colocan las bases o fundamentos de todas las etapas posteriores de esta metodología.

Análisis de requerimientos

Esta etapa proporciona información del contexto del problema para identificar y establecer el comportamiento que va a tener el sistema y delimitar su funcionalidad.

Respecto a la red social de aprendizaje, las funcionalidades definidas coinciden con los de sistemas similares (redes sociales); para la herramienta Fermat, estas son:

- ✓ Cuentas de usuario.
- ✓ Comunidades.
- ✓ Cursos.
- ✓ Enlaces de amistades.
- ✓ Envío de mensajes.

Estas funcionalidades se mapean en un conjunto de requisitos que se clasifican en funcionales y no funcionales. En la tabla 6 se muestra la lista principal de requisitos funcionales de la red social de aprendizaje Fermat.

Tabla 6. *Requisitos funcionales de la red social de Fermat*

<i>ID</i>	<i>Requisito</i>
R1	El usuario debe poder crear cuentas de usuario
R2	Los usuarios podrán iniciar y finalizar sesión
R3	Los usuarios podrán agruparse en comunidades
R4	Los usuarios podrán enviar, recibir, eliminar y tener un historial de mensajes
R5	Los usuarios podrán tomar cursos y exámenes
R6	Fermat deberá almacenar el historial de calificaciones y cursos de los estudiantes
R7	Los usuarios podrán invitar, agregar y eliminar amigos
R8	Los usuarios podrán conocer la ubicación actual de ellos y sus amigos

Respecto a los sistemas tutores inteligentes, la tabla 7 expone algunos de los requisitos establecidos que deben estar presentes.

Tabla 7. *Requisitos del sistema tutor inteligente*

<i>ID</i>	<i>Requisito</i>
R9	Procesar de manera secuencial la resolución de operaciones de multiplicación y división
R10	Identificar errores a través de las casillas de entrada
R11	Proporcionar ayuda dependiendo de la etapa de resolución de la operación
R12	Medir el tiempo de resolución de la operación de manera interna
R13	Adaptar la interfaz del tutor de acuerdo con el estilo de la red Fermat
R14	Integrar al tutor inteligente un sistema experto difuso para que ajuste la dificultad en las operaciones de acuerdo con el desempeño del estudiante
R15	Revisar en un determinado tiempo si la entrada del estudiante es correcta

Una vez reunidos los requisitos funcionales de la red social y del sistema tutor inteligente, se desarrollan los escenarios de interacción entre los usuarios y el sistema, llamados “casos de uso”. Los casos de uso ayudan a representar las funcionalidades del sistema que están implícita o explícitamente en los requerimientos obtenidos en la actividad del análisis. Algunos de los casos de usos mapeados a requisitos funcionales se muestran en la tabla 8.

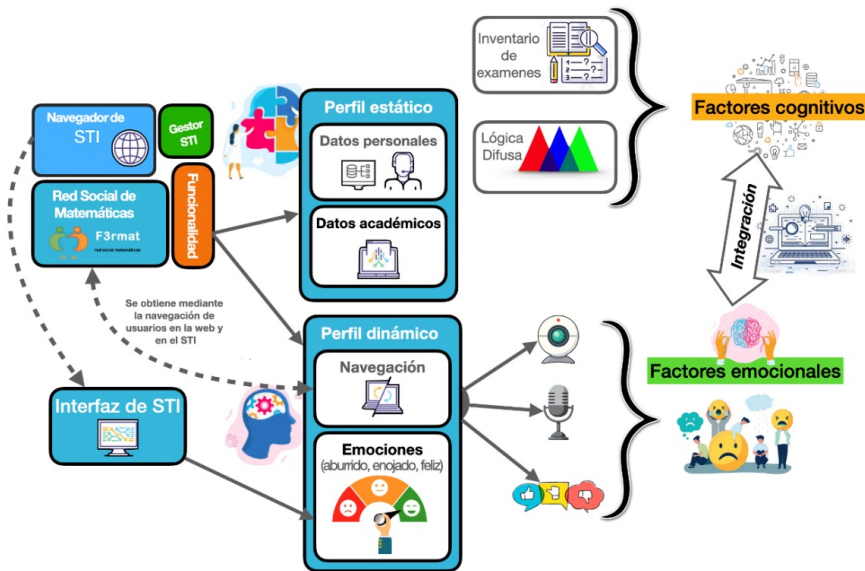
Tabla 8. *Mapeo de Casos de Uso de Fermat a requisitos funcionales*

<i>ID</i>	<i>Caso de uso</i>	<i>R1</i>	<i>R2</i>	<i>R11</i>	<i>R12</i>	<i>R14</i>
CU1	Registrar usuario	X				
CU2	Iniciar sesión		X			
CU15	Proporcionar ayuda			X		
CU16	Cambiar dificultad en problema				X	X

Diseño de Fermat

Una vez detectados los principales requerimientos o requisitos de todo el *software*, se procedió a realizar el diseño. La figura 40 muestra un esquema general de Fermat, incluyendo la red social y los dos sistemas tutores inteligentes.

Figura 40. Esquema de los elementos de Fermat



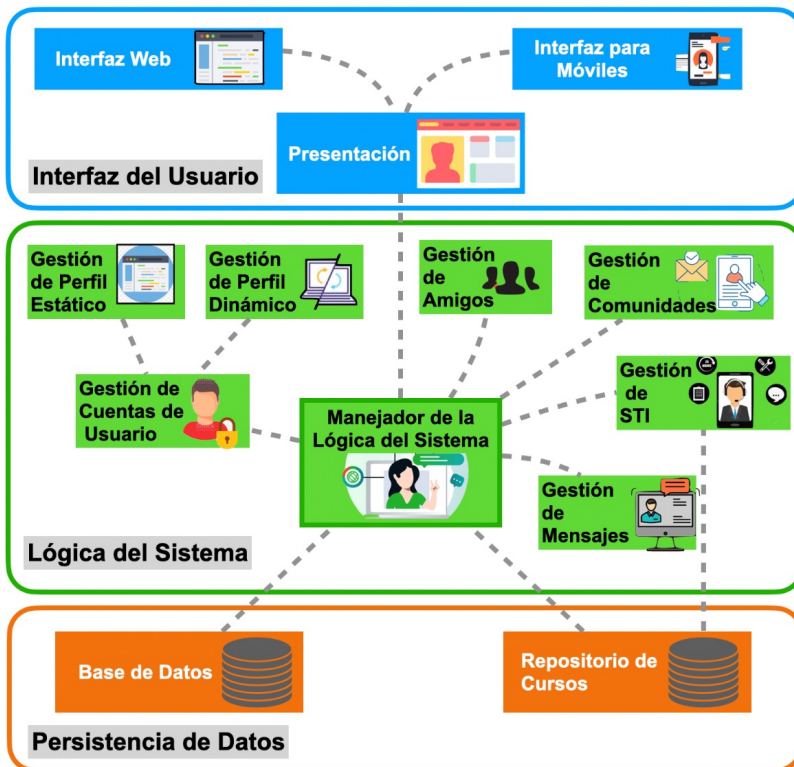
Como se puede apreciar en la figura 40, un usuario accede a la red social mediante un navegador como Explorer, Firefox o Chrome. El usuario cuenta con dos perfiles: uno estático, el cual contiene datos personales (nombre, correo, edad, etc.) y académicos (escuela, cursos, calificaciones, etc.), y otro dinámico, el cual mantiene información en tiempo real de aspectos emocionales (aburrido, frustrado, motivado, etc.) y cognitivos (resultados, tiempo, y errores en ejercicios). Los datos almacenados en estos perfiles se captan durante la interacción del usuario con el sistema de distintas fuentes. Los datos del perfil estático se obtienen cuando el usuario se registra a la red social o cuando resuelve el examen de valorización de matemáticas. Los datos del perfil dinámico se obtienen de exámenes en línea, como el Felder-Soloman (Felder & Soloman, 2004), de ejercicios (tiempo, ayudas y

fallas) y desde dispositivos como la cámara web y el micrófono. Con estos datos (emociones, tiempos, errores o fallas, etc.) un sistema difuso toma la decisión del grado de complejidad del siguiente ejercicio.

Una vez ingresado a la red social, el usuario puede acceder a los sistemas tutores inteligentes, como se aprecia en la misma figura 40. El tutor cuenta con su propia interfaz para que el estudiante ejercite distintas operaciones de multiplicación y división en niveles que dependerán de su habilidad para resolver los ejercicios (nivel de conocimiento). Este nivel se asigna al inicio por medio de un examen de conocimientos, pero después se calcula dinámicamente por el sistema difuso que contiene el tutor.

Los diferentes módulos de la red social se distribuyeron para conformar la arquitectura de software del sistema que se presenta en la figura 41.

Figura 41. Arquitectura de software para Fermat



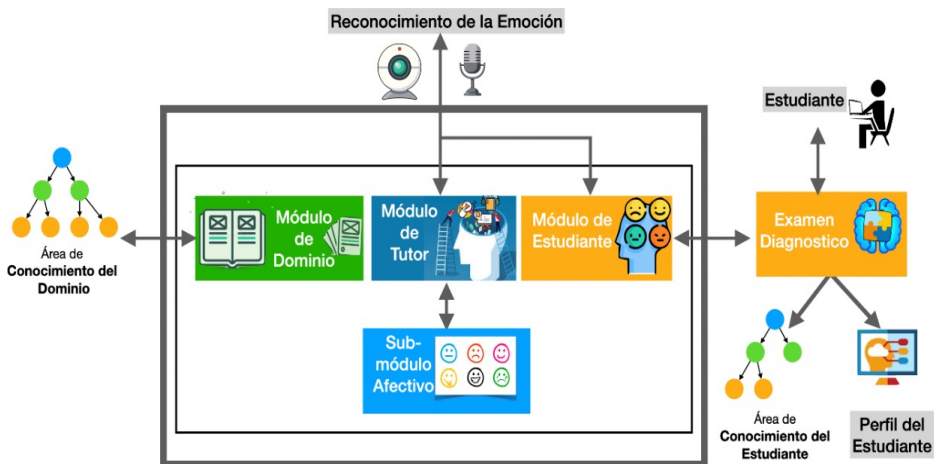
Se utilizó el modelo arquitectónico de capas de abstracción con tres capas: Interfaz del Usuario, Lógica del Sistema y Persistencia de Datos. Este diseño arquitectónico en capas permite dividir los componentes del sistema en diferentes niveles de abstracción y desarrollar de forma independiente cada uno de ellos. Las capas sólo pueden interactuar con sus capas adyacentes, lo que permite mantener la consistencia en los datos; además, mientras se respeten las interfaces de comunicación entre capas, los cambios realizados en una capa no afectan a las demás.

Se propuso este estilo arquitectónico debido a su soporte para adecuarse a los cambios, para que las futuras modificaciones a la red social puedan desarrollarse sin afectar a los demás componentes; además, permite tener un mejor control sobre la persistencia de los datos y la seguridad de estos.

Estructura del sistema tutor inteligente

La estructura del sistema tutor inteligente (STI) respeta el modelo tradicional de un STI, el cual contiene cuatro componentes: una interfaz de usuario encargada de la comunicación entre el usuario y los otros tres componentes llamados módulos, que son: dominio, estudiante y tutor. El sistema también cuenta con un módulo que realiza el reconocimiento de emociones y las comunica al sistema. La figura 42 muestra la estructura completa del STI de Fermat.

Figura 42. Estructura del STI de Fermat



A continuación, se describe la función e implementación de cada uno de los módulos.

Módulo de Dominio: Este módulo contiene todo el conocimiento del dominio del experto, en este caso correspondiente a los métodos de solución de multiplicaciones y divisiones de números enteros que se enseñan en tercer año de primaria. El conocimiento se implementó como un árbol en donde la raíz representa el conocimiento total de la operación (por ejemplo, la operación división), sus ramas representan los capítulos y las hojas vienen a ser los temas. La totalidad de los nodos en el árbol de conocimiento representa el conocimiento del módulo de dominio (experto).

Los temas que conforman al curso se encuentran almacenados en un archivo en formato XML con diferentes etiquetas para cada elemento del curso:

```
<curso> </curso> delimitan el contenido del curso  
<nombre></nombre> define el nombre del curso que se creó  
<temas></temas> delimita el contenido del curso por temas.
```

Para cada tema se definieron tres etiquetas como se muestra a continuación:

```
<tema id = "número"> </tema> define el número de tema del curso  
<urlCurso></urlCurso> define la dirección URL del tema del curso  
<urlExamen></urlExamen> define la dirección URL del examen de un tema del curso.
```

Un ejemplo de la estructura del archivo XML del curso con las etiquetas mencionadas se muestra a continuación:

```
<?xml version="1.0" encoding="UTF-8"?>
<curso>
  <nombre>Nombre del curso</nombre>
  <temas>
    <tema id="1">Tema1 </tema>
    <urlCurso>Curso/Tema1.xml</urlCurso>
    <urlExamen>Curso/Examen_Tema1.xml</urlExamen>

    <tema id="2">Tema2 </tema>
    <urlCurso>Curso/Tema2.xml</urlCurso>
    <urlExamen>Curso/Examen_Tema2.xml</urlExamen>

    <tema id="3">Tema3 </tema>
    <urlCurso>Curso/Tema3.xml</urlCurso>
    <urlExamen>Curso/Examen_Tema3.xml</urlExamen>
  </temas>
</curso>
```

Cada tema tiene asociada la URL donde está almacenado el contenido del tutor inteligente de cada tema del curso y el examen de diagnóstico para conformar, a través del módulo del estudiante, sus perfiles.

Módulo de Estudiante: El módulo del estudiante de Fermat contiene toda la información del estudiante. Por una parte, se representa el conocimiento que tiene el estudiante, el cual puede ser visto como un subárbol de todo el conocimiento que el experto posee (área de conocimiento del estudiante), y por otra contiene también la información de los perfiles (estático y dinámico) del estudiante. El módulo del estudiante aplica un examen diagnóstico al estudiante la primera vez que accede al STI con el fin de evaluar el desempeño del estudiante y determinar sus habilidades cognitivas y de razonamiento. Los resultados del examen muestran lo que el estudiante sabe y lo que necesita aprender, y esta información se representa en el árbol de conocimiento (dominio) del estudiante. Toda la información personal y académica del estudiante se almacena en el perfil del estudiante.

Para determinar el grado o nivel de conocimientos del estudiante, se implementó un examen diagnóstico. El contenido del examen se almacena en un archivo XML con la siguiente estructura:

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<xml>
  <Examen>
    <Titulo>Título del Examen</Titulo>
    <Descripcion>Instrucciones </Descripcion>
    <Tema> Tema asociado </Tema>
    <Preguntas>
      <Pregunta dificultad="basico" id="p1">Pregunta 1
        <Respuestas>
          <Respuesta correcto="false">r1</Respuesta>
          <Respuesta correcto="true">r2</Respuesta>
          <Respuesta correcto="false">r3</Respuesta>
          <Respuesta correcto="false">r4</Respuesta>
        </Respuestas>
      </Pregunta>
      <Pregunta dificultad="normal" id="p2">Pregunta 2
        <Respuestas>
          <Respuesta correcto="true">r1</Respuesta>
          <Respuesta correcto="false">r2</Respuesta>
          <Respuesta correcto="false">r3</Respuesta>
          <Respuesta correcto="false">r4</Respuesta>
        </Respuestas>
      </Pregunta>
      <Pregunta dificultad="dificil" id="p2">Pregunta 3
        <Respuestas>
          <Respuesta correcto="true">r1</Respuesta>
          <Respuesta correcto="false">r2</Respuesta>
          <Respuesta correcto="false">r3</Respuesta>
          <Respuesta correcto="false">r4</Respuesta>
        </Respuestas>
      </Pregunta>
    </Preguntas>
  </Examen>
</xml>
```

Cada pregunta tiene asignado un nivel de dificultad que se usa para determinar la ponderación de cada respuesta en la evaluación. En este caso, las preguntas difíciles tienen un valor de 3 puntos, las normales de 2 y las básicas de 1 punto.

Para la calificación (valor entre 0 y 1), se utiliza la siguiente ecuación:

$$\text{Calificación} = \frac{\sum_{i=1}^n r_i}{\sum_{i=1}^n p_i}$$

donde:

r_i representa los puntos de cada respuesta correcta del examen,

p_i representa los puntos asignados a cada pregunta del examen.

El sistema determina el resultado del examen, y con éste un algoritmo obtiene el nivel de aprendizaje del estudiante y el método de enseñanza que se debe utilizar.

El algoritmo para asignar el nivel y método (“Lattice” o tradicional) para el aprendizaje del estudiante en la operación de multiplicación, se presenta a continuación.

```

Si (Calificación < .4) Entonces
  Nivel = Fácil;
  Método = Lattice
SiNo
  Si (Calificación < .5) Entonces
    Nivel = Fácil;
    Método = Tradicional
  SiNo
    Si (Calificación < .9) Entonces
      Nivel = Normal;
      Método = Tradicional
    SiNo
      Nivel = Difícil;
      Método = Tradicional
  FinSi
FinSi
FinSi
FinSi

```

Para la representación del conocimiento del estudiante se usaron dos categorías:

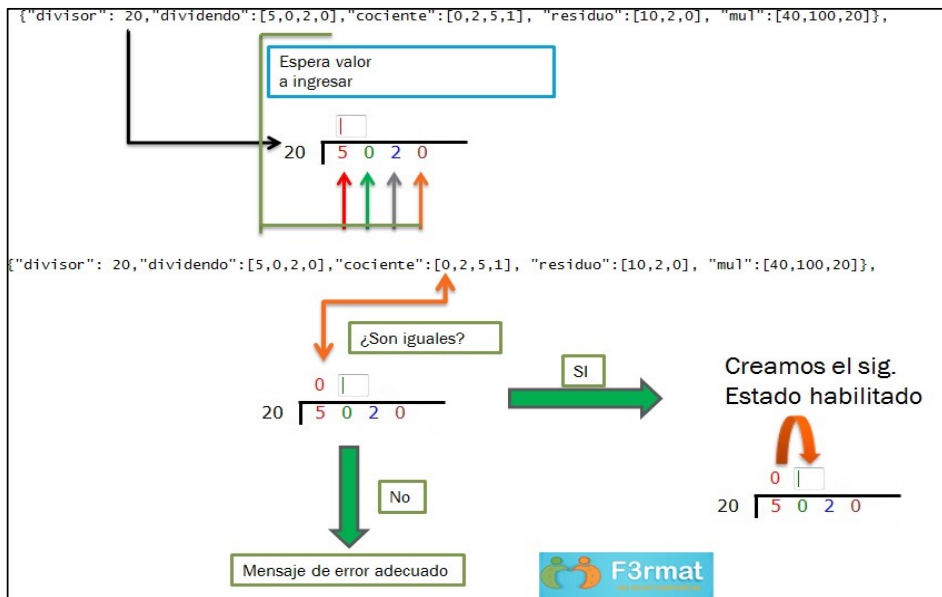
- **Temas.** Cada vez que el estudiante accede a un tema, este se almacena en el historial de temas visitados por el estudiante.
- **Experiencia del estudiante.** Almacena el historial de las calificaciones obtenidas en los exámenes que el estudiante ha respondido por materia.

Módulo de Tutor: El módulo tutor del STI está construido siguiendo dos modelos. El primero está basado en la teoría de cognición de ACT-R (Anderson *et al.*, 1990). Este tipo de sistemas tutores son llamados “tutores de modelo-seguimiento”, donde al estudiante se le enseña cada acción que debe realizar paso a paso. El segundo modelo se apega más al módulo del estudiante, y muestra los posibles errores que se pueden cometer durante la solución de un ejercicio (Woolf, 2009). Estos modelos se eligieron porque se adaptan al proceso que se sigue para resolver una división de números enteros. Enseñar a resolver divisiones es un proceso que el estudiante debe conocer paso a paso, por lo que indicar los errores que se cometen durante el proceso es importante para que el estudiante aprenda de ellos. El módulo del tutor cuenta con el apoyo de dos submódulos adicionales:

- **Reconocimiento de emociones:** Las emociones se detectan utilizando las expresiones faciales y la voz. El método usado para la detección de emociones en rostro está basado en la teoría de Ekman, que distingue seis emociones básicas (ira, disgusto, miedo, felicidad, tristeza y sorpresa), a los cuales se agregó una más para representar un estado neutral.
- **Manejo afectivo:** La instancia de este submódulo es un agente afectivo que emite mensajes que ayudan al estudiante a sentirse acompañado, indica los errores que comete durante el curso y ofrece ayuda cuando el estudiante la solicita. Este agente animado se representa por el personaje llamado “Genio” que proviene del grupo de agentes implementados por Microsoft.

El módulo tutor fue implementado mediante una interfaz en HTML5 y JavaScript, y en él se reciben las respuestas del estudiante y se comprueban mediante un intercambio de datos a través de objetos XML para las multiplicaciones y JSON para las divisiones. La figura 43 muestra el mecanismo de evaluación del tutor para una operación de división. En la figura se puede observar que aparecen dos líneas de texto donde se despliega la información necesaria para verificar las respuestas del estudiante en cada paso y proporcionar ayuda, o bien habilitar el siguiente estado.

Figura 43. Mecanismo de evaluación del tutor

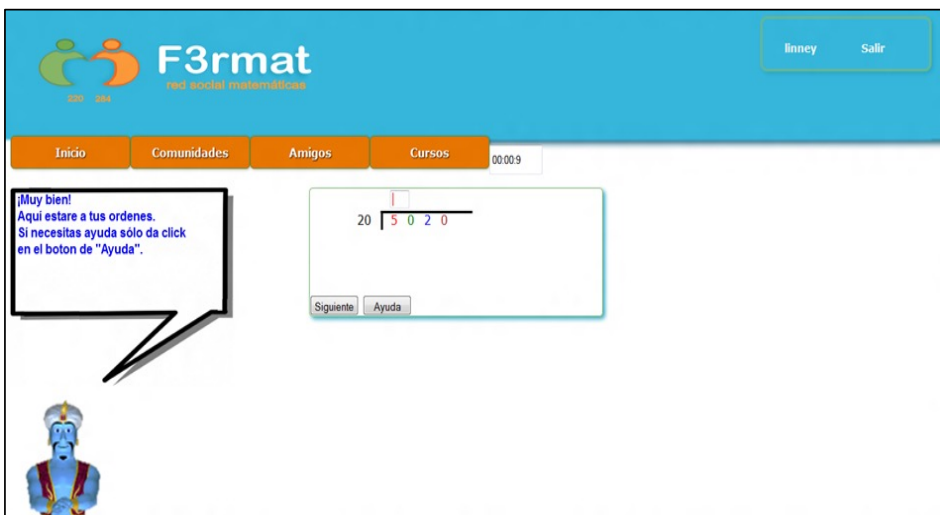


En este sistema se puede observar que el tutor espera que el usuario ingrese un valor usando la interfaz; posteriormente, el sistema verifica que sea correcto y activa la siguiente casilla para continuar el proceso con el siguiente valor. Cuando el usuario ingresa un valor erróneo, el sistema tutor utiliza al agente afectivo para enviar un mensaje sobre el tipo de error. Este flujo de interacción se repite hasta finalizar el proceso de la división. Durante este proceso el estudiante puede hacer uso de dos botones que se encuentran debajo del espacio de la operación de división que está resolviendo. El botón de Ayuda envía mensajes de retroalimentación al estudian-

te por medio del agente afectivo. El botón Siguiente tiene dos funciones principales: cambia de operación avanzando a la siguiente, independientemente de si la división actual está terminada o no, y también llama a ejecución al sistema experto difuso.

La figura 44 muestra la interfaz real que se presenta en FERMAT para resolver operaciones de división. En la interfaz se puede apreciar al agente pedagógico Genio y un mensaje de retroalimentación, además se muestran también los botones de Ayuda y Siguiente. La figura del agente afectivo cambia de acuerdo con su respuesta, la cual busca ser cordial o afectuosa con el estudiante. Genio aparece cuando el estudiante comete un error o cuando este solicita ayuda.

Figura 44. Interfaz del tutor inteligente FERMAT

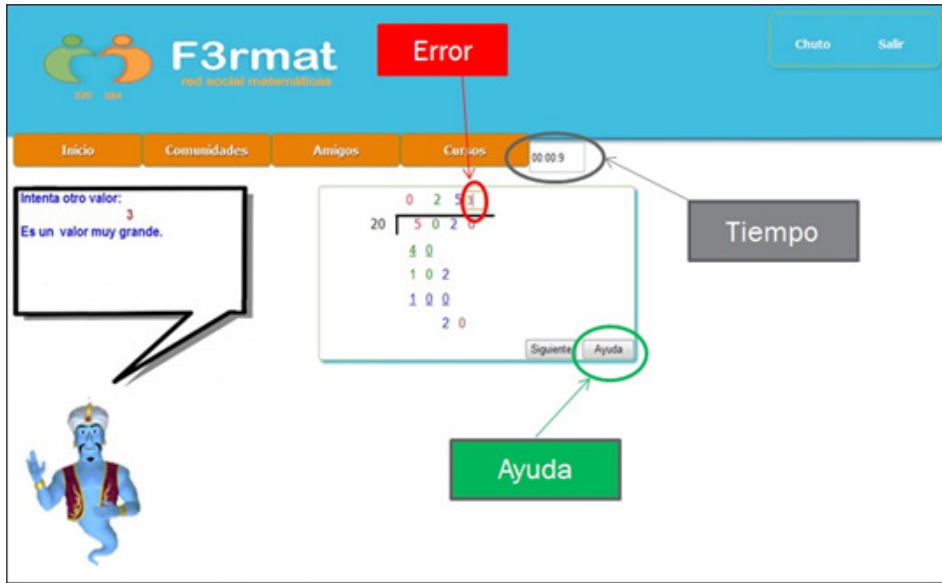


7.4 Sistema experto difuso

Para realizar el trabajo de razonamiento o inferencia de los ejercicios que se presentan al estudiante de acuerdo con su nivel de conocimientos, se implementó un sistema experto difuso. Este sistema toma como entrada tres variables difusas que tienen que ver con el tiempo, el número de errores y el número de ayudas, las cuales se determinan en el ejercicio que se está

resolviendo y con ellas genera un valor difuso de salida que es la dificultad del siguiente ejercicio. En cierta forma, el rendimiento del estudiante se ve reflejado mediante esas variables al estar resolviendo una operación. Las tres variables difusas de entrada mencionadas se señalan en la figura 45.

Figura 45. Las tres variables de entrada del sistema difuso

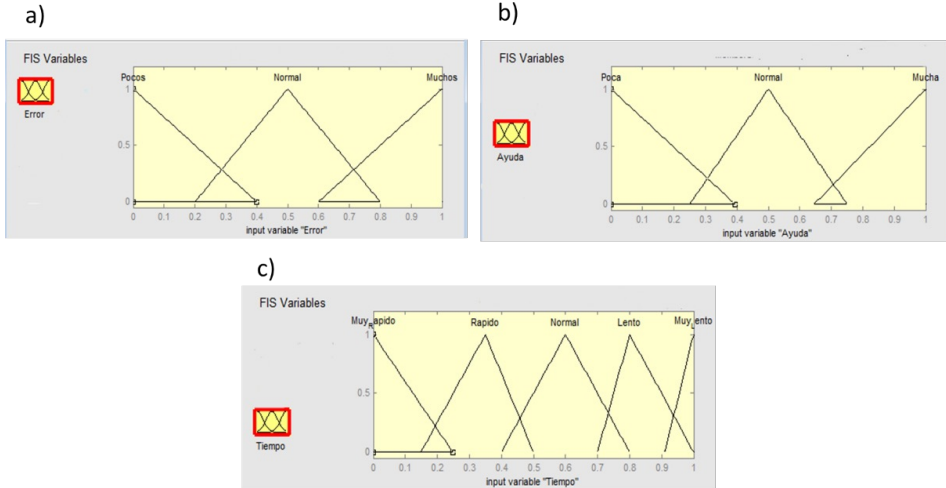


Los conjuntos difusos propuestos para cada variable lingüística son los siguientes:

- **Error** = {poco, normal, muchos}
- **Ayuda** = {poca, normal, mucha}
- **Tiempo** = {muy rápido, rápido, lento, muy lento}
- **Dificultad** = {muy fácil, fácil, básico, difícil, muy difícil}

En la figura 46 se muestran gráficamente los conjuntos difusos de cada una de las variables lingüísticas Error, Ayuda y Tiempo.

Figura 46. Conjuntos difusos de variables Error, Ayuda y Tiempo



Algunas reglas difusas definidas para determinar el grado de dificultad del siguiente ejercicio se muestran a continuación:

Si (Error es poco) &
 (Ayuda es poca) &
 (Tiempo es muy rápido) entonces
 (Dificultad es muy difícil)

Si (Error es poco) &
 (Ayuda es normal) &
 (Tiempo es lento) entonces
 (Dificultad es difícil)

Si (Error es muchos) &
 (Ayuda es mucha) &
 (Tiempo es muy lento) entonces
 (Dificultad es muy fácil)

La tabla 9 muestra los valores difusos, y sus respectivos rangos, y valores normalizados para la variable difusa Dificultad.

Tabla 9. Valores difusos para variable Dificultad

Variable Difusa	Rango de dificultad (%)	Valores normalizados
Muy fácil	0% - 10%	0 - 0.1
Fácil	0% - 30%	0 - 0.3
Normal	20% - 80%	0.2 - 0.8
Difícil	70% - 100%	0.7 - 1.0
Muy difícil	90% - 100%	1.0

7.5 Reconocimiento de emociones por medio de redes neuronales

El reconocimiento de afecto o emociones es una de las áreas más interesantes y prometedoras en el campo de los sistemas tutores inteligentes. La instrucción adaptada o personalizada originalmente tenía como principal objetivo usar la psicología y el análisis cognitivo para crear STI basados en el uso de computadoras (Carbonell, 1970; Clancey, 1979; Anderson *et al.*, 1990; Alevén & Koedinger, 2002; Woolf, 2009). Sin embargo, Picard (1995) argumenta que, al observar las emociones del estudiante, un sistema computacional que enseña (sistema tutor) podría responder como lo haría un tutor humano, es decir, personalizando la enseñanza.

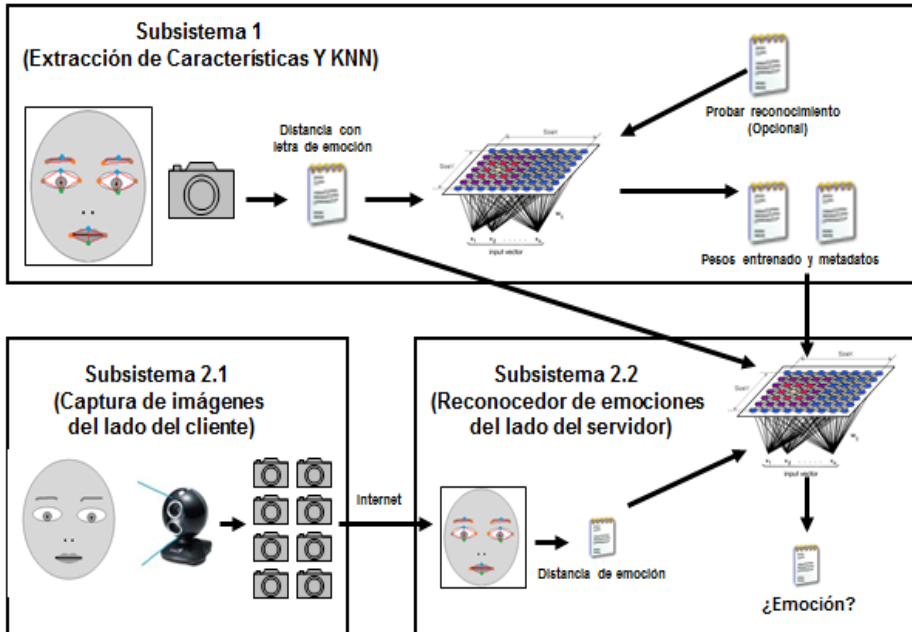
En los últimos años se ha observado un aumento en el número de STI sensitivos al afecto aplicados a diferentes campos de aprendizaje (Hasan *et al.*, 2020; Petrovica *et al.*, 2017). Un sistema que detecta emociones o afectos observa y estudia la cara, la voz, la conversación y otros elementos o características de un usuario, para enseguida detectar frustración, interés, aburrimiento u otras emociones (Essa & Pentland, 1995; Yacoob & Davis, 1996; Tosa & Nakatsu, 1996; Graesser *et al.*, 2004). Por otra parte, un sistema responde a emociones buscando que el usuario transite de un estado emocional negativo a uno positivo.

Uso de una red neuronal para reconocimiento de emociones

En Fermat las emociones se detectan por medio de la cara y a través de la voz y la conversación o diálogo. El algoritmo usado para detectar emociones en rostros se basa en la teoría de Ekman, la cual reconoce siete emociones básicas: enojo, disgusto, miedo, felicidad, tristeza, sorpresa y neutro (carencia de emoción). El sistema reconocedor fue construido en tres subsistemas: el primero fue implementado para extraer características de rostros almacenados en una base de datos o *dataset* para ser usadas en el entrenamiento de la red neuronal. El segundo subsistema es la red neuronal de Kohonen, que se usa para reconocer emociones en rostros. Y el tercer subsistema integra a los dos primeros en un modelo cliente (extracción de características) y ser-

vidor (red neuronal para reconocer emociones). La figura 47 muestra todo el sistema de reconocimiento de emociones en rostro, el cual fue implementado en el lenguaje de programación Java con ayuda de las librerías OpenCV (Open Source Computer Vision) y JavaCV.

Figura 47. Sistema reconocedor de emociones



A continuación, se explica el algoritmo usado en el sistema.

Algoritmo de extracción de características

- I. *Habilitar la captura de video* a través de la cámara web.
- II. *Detección de la cara.* Aplicar la técnica Haar-like Features Cascades (Viola & Jones, 2001; Lienhart & Maydt, 2002), para la detección de la cara de la persona. La detección creada por este método permite la creación de zonas límites en la imagen. Estas zonas son conocidas como "regiones de interés" (ROI por sus siglas en inglés). Las ROI tienen correspondencia con las partes de donde se piensa extraer las características y medidas que serán la entrada a la red neu-

ronal. Una vez detectada la cara y delimitada las ROI, se procede a dibujar un rectángulo alrededor de ella.

- III. *Detección de la boca.* Aplicar Haar-like Features Cascades a la ROI creada en el paso anterior, para la detección de la boca de la persona y dibujar un rectángulo alrededor de la ROI. La imagen delimitada por la ROI se convierte a escala de grises. Se le aplica una ecualización de histograma a la imagen delimitada en la ROI que permite crear una media de la iluminación de la imagen.
- IV. *Establecer límites.* Se le aplica un *threshold* a la imagen delimitada en la ROI. El proceso de *threshold* permite la estipulación de un número, el cual funciona como umbral; en caso de que este umbral sea superado por la cantidad de gris de un pixel dado, el pixel en cuestión se volverá negro, en caso contrario se volverá blanco. Este proceso ocasiona que la imagen se reduzca a solo dos colores, negro y blanco. La imagen delimitada en la ROI se guarda en una matriz del tamaño de la imagen en pixeles.
- V. *Coordenadas alrededor de la imagen.* En este paso se escanea la imagen y se localizan los pixeles negros en las posiciones más bajas, más a la izquierda, derecha y más altas. Con esto se obtienen cuatro coordenadas (x, y) alrededor de la imagen.
- VI. *Detectar ojos.* Se aplica Haar-like Features Cascades para detectar los ojos derecho e izquierdo de la persona. Se dibuja un rectángulo alrededor de la nueva ROI y se aplican los pasos *iii* y *iv* para delimitar los ojos. Respecto a la ceja, se aplica el mismo procedimiento, pero solo se obtienen tres coordenadas.
- VII. *Dibujar líneas entre coordenadas.* Se dibujan las líneas entre las coordenadas establecidas para cada ROI. Estas 22 líneas representan “distancias” que se utilizan para detectar la emoción de la persona (véase la figura 48).
- VIII. *Para cada una de las 22 distancias, se aplica el teorema de Pitágoras* entre las dos coordenadas de cada distancia y se guarda el resultado de la hipotenusa del triángulo. En caso de que el teorema de Pitágoras no se pueda aplicar, ya sea porque los dos puntos están alineados vertical u horizontalmente, se guarda la diferencia positiva entre las dos coordenadas que no son iguales.

IX. *Normalización de valores.* Cada una de las 22 distancias se divide por la anchura de la cara. Esta división normaliza los valores de distancia en el intervalo 0 a 1.

X. *Asignar la emoción.* La emoción es el resultado deseado de la red neuronal al usar la expresión del rostro que se representa por una letra de acuerdo con los valores calculados previamente.

XI. *Almacenar información.* Se crea y guarda una nueva línea en un archivo de texto con la letra que representa la emoción y las 22 distancias. La figura 49 muestra una parte del archivo donde cada línea representa las características extraídas de un rostro.

XII. *Finaliza la extracción de características del rostro actual y se repite el proceso con un nuevo rostro o cara.*

Figura 48. Rostro con las 22 distancias usadas para reconocer emociones

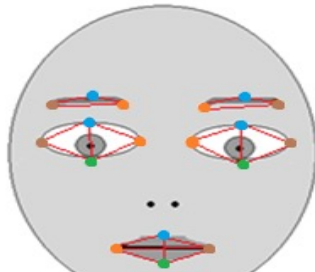
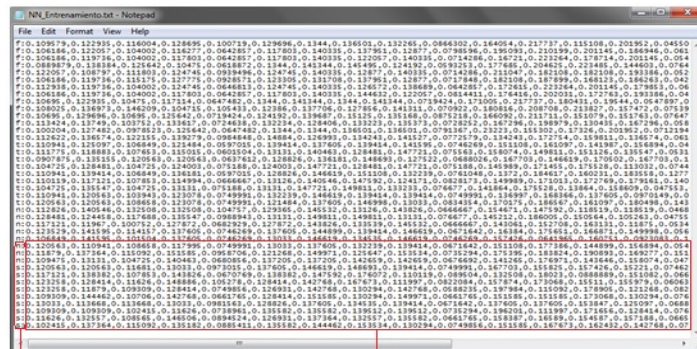


Figura 49. Archivo con emociones y distancias para entrenar a la red neuronal



Letra que representa la emoción

22 puntos de medida de la emoción

Entrenamiento y prueba de la red neuronal de Kohonen

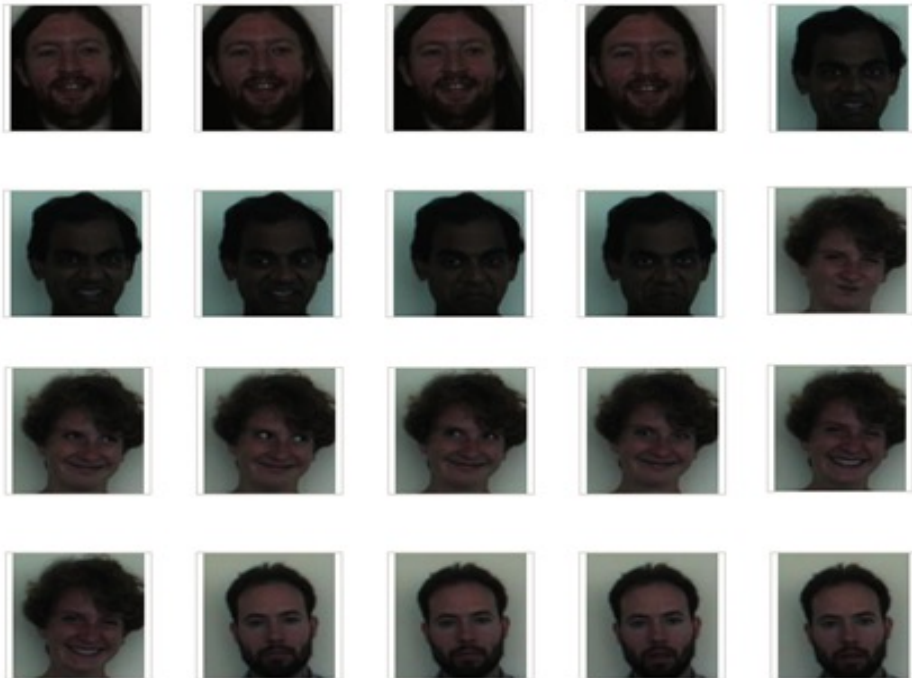
La red neuronal fue entrenada con diferentes conjuntos de datos de caras. En el proceso de entrenamiento y con los resultados obtenidos, se procedió a realizar cambios en la extracción de características y en el algoritmo de entrenamiento de la red. A continuación, se explica cómo se realizó este proceso con diferentes conjuntos de datos.

Conjunto de datos Grimace

(<https://cmp.felk.cvut.cz/~spacelib/faces/grimace.html>)

Contiene un total de 360 imágenes, las cuales se dividen en 18 modelos, cada una con 20 imágenes. Este conjunto de datos presenta a un grupo de individuos, los cuales hacen muecas que son capturadas en fotos. Un ejemplo de estas imágenes se presenta en la figura 50. Dos desventajas de Grimace son que las imágenes no están organizadas por emociones y que tienen una mala iluminación.

Figura 50. Ejemplo de imágenes de la base de datos Grimace



Conjunto de datos JAFFE

(https://www.kasrl.org/jaffe_download.html)

Japanese Female Facial Expression es un conjunto de datos de expresiones faciales que contiene 213 imágenes. Estas se dividen en siete expresiones faciales simuladas por diez modelos japonesas femeninas. Las expresiones faciales son: sorpresa, enojo, miedo, disgusto, felicidad, tristeza y neutral. En la figura 51 se observa un ejemplo de estas imágenes. Las desventajas de este conjunto de datos son que solo cuenta con rostros femeninos, que pertenecen a solo una raza (poca variedad), y que están en blanco y negro.

Figura 51. Ejemplo de una modelo mostrando cinco emociones en JAFFE



Conjunto de datos RaFD

La Radboud Faces Database (Langner *et al.*, 2010) es una base de datos de expresiones faciales pertenecientes a un conjunto de 67 modelos que incluye hombres y mujeres caucásicos, niños y niñas caucásicos y hombres marroquí-holandeses. Los modelos muestran ocho expresiones emocionales, las cuales corresponden al FACS de Ekman (Ekman & Freisen, 1978). Un ejemplo de esta base de datos se muestra en la figura 52. La RaFD fue la base de datos que dio mejores resultados para Fermat.

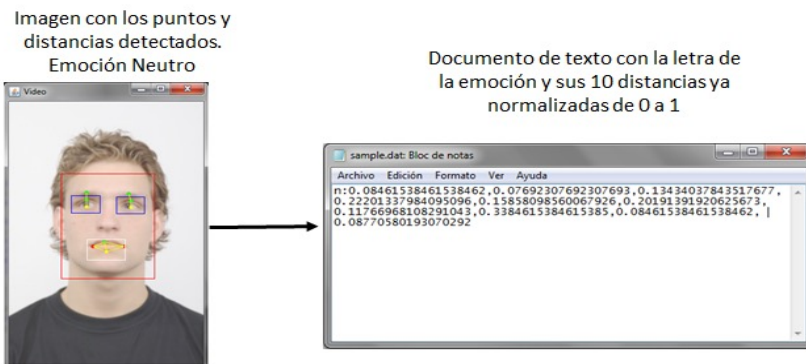
Figura 52. Ejemplo de un modelo mostrando ocho emociones en RaFD



Ejemplo del extractor de características y el entrenamiento de la red neuronal

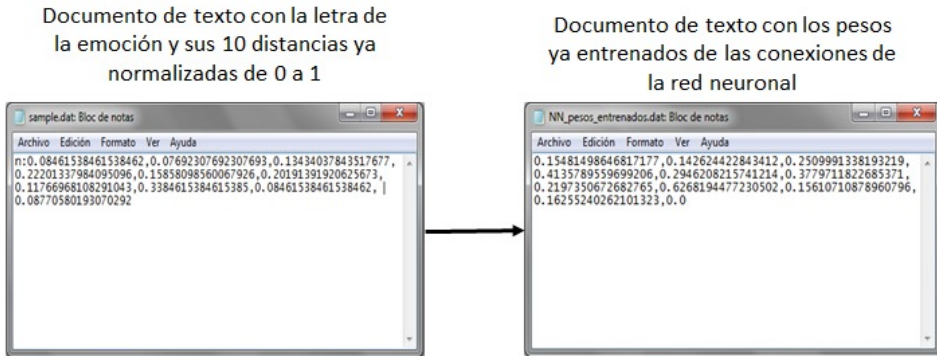
A continuación se describe un ejemplo para mostrar la forma en que el extractor de características captura las distancias y las normaliza. Posteriormente se muestra el procesamiento de la red neuronal. Para el ejemplo se tomó una imagen de la base de datos RaFD (véase figura 53) cuya emoción reconocida es la de neutro (letra **n**). Primero se señalan los puntos de la cara y se calculan las distancias. Después se normalizan los valores y se guardan en el archivo de texto. En este ejemplo solo aparecen diez distancias almacenadas, que son el resultado de los últimos cambios realizados al algoritmo de extracción de características.

Figura 53. Conversión y normalización de las distancias



Finalmente se muestra la conversión de las distancias ya normalizadas con los pesos finales de las conexiones generados por la red neuronal (figura 54).

Figura 54. Conversión de las distancias normalizadas a pesos ya entrenados



Tecnologías usadas en el desarrollo

La red social FERMAT y los sistemas tutores inteligente para el aprendizaje de operaciones de multiplicación y división de números enteros fueron implementados con diferentes herramientas de *software* y lenguajes de programación.

La red social FERMAT fue implementada usando un modelo arquitectónico de tres capas; en el desarrollo de cada capa se usaron diferentes tecnologías.

La capa de presentación, que es responsable de interactuar con el usuario a través de diferentes dispositivos móviles y de escritorio, se implementó usando el lenguaje de programación JavaScript, el lenguaje de marcas HTML5 y el lenguaje de diseño de hojas de cascada CSS3, permitiendo así la adecuación dinámica del contenido que se presenta al usuario a través de la pantalla del dispositivo que usa.

Por otra parte, la capa lógica del sistema contiene los módulos para gestionar la red social, el sistema tutor inteligente encargado de enseñar las operaciones de multiplicación y división y de reconocer las emociones del usuario. Los componentes de esta capa lógica se desarrollaron usando el lenguaje de programación Java y JSP para la creación de páginas web diná-

micadas basadas en el lenguaje HTML y XML para el intercambio de datos. El módulo de gestión del STI contiene la lógica del sistema tutor inteligente y además realiza el reconocimiento de emociones a través de expresiones del rostro captadas en imágenes. El sistema personaliza la instrucción usando en el sistema difuso la información detectada durante la sesión de aprendizaje. La información que se usa en el sistema difuso para determinar el grado de complejidad del siguiente ejercicio consiste en 1) el estado emocional del estudiante, 2) el tiempo invertido en un ejercicio, 3) el número de errores cometidos en la solución de un ejercicio y 4) el número de ayudas o apoyos pedagógicos solicitados por el estudiante durante la sesión.

Finalmente, la capa de datos se implementó a través del manejador de base de datos MySQL y el lenguaje de marcas XML, con el fin de resguardar la información necesaria que almacena los cursos, ejercicios, perfiles de estudiantes, etc., los cuales son indispensables para el correcto funcionamiento del sistema.

8. Memorato: juego clásico en una época moderna

Actualmente existen muchas aplicaciones de *software* y juegos en línea sobre una amplia variedad de temas, incluidos, por supuesto, temas que se abordan en ambientes escolares como parte de los programas educativos de formación de estudiantes en diversos niveles educativos. Sin embargo, un docente enfrenta muchos retos al intentar incluir una actividad o aplicación en su práctica diaria que le permita mostrar contenido y ejercicios a sus estudiantes con el objetivo de reforzar las actividades de aprendizaje. En este contexto, el uso de *software* para reforzar el aprendizaje de los estudiantes tiene como objetivo incrementar la motivación de los estudiantes con el fin de mejorar su concentración para lograr aprendizajes más significativos y duraderos al estudiar o repasar el contenido con aplicaciones de software. La atención de los estudiantes es indispensable durante el proceso de aprendizaje; hoy en día es innegable que los jóvenes se sienten atraídos por el uso de la tecnología (celulares, tabletas, consolas de videojuegos, etc.) para realizar muchas de sus actividades cotidianas. Desde hace tiempo se formuló una pregunta común en el sector educativo que sigue vigente en esta época: ¿de qué manera se puede integrar la tecnología a una estrategia que promueva el aprendizaje?

Por otro lado, el docente debe conocer la evolución de cada uno de sus estudiantes y desarrollar actividades que promuevan su motivación. Desafortunadamente, si un docente busca una aplicación para cada tipo de ejercicio que desarrolla en cada tema, podría ocurrir que no encuentre o no se

sienta satisfecho con el *software* de reforzamiento que esté disponible. Las razones de esta insatisfacción se pueden deber a diferentes aspectos como que el software no es gratuito, que los ejercicios que contiene son de nivel muy básico o más alto del requerido en ese momento. Por ejemplo, si un docente está comenzando a enseñar la operación de adición de enteros de un dígito, con números en el rango del 1 al 5, quizá logre encontrar un software que sea gratuito y que tenga ejercicios semejantes a los que los estudiantes requieren para practicar.

La gamificación juega un papel muy importante dentro del proceso educativo (Marín & Hierro, 2013). La también llamada ludificación se refiere al uso de estrategias de juego para alcanzar un objetivo: el aprendizaje. La principal ventaja es incrementar la motivación de los estudiantes en un ambiente lúdico, competitivo y bajo la supervisión del docente.

8.1 ¿Qué es Memorato?

Memorato (Saucedo *et al.*, 2020) es una plataforma basada en el juego del memorama o juego de memoria. Esta plataforma permite a un docente crear fácilmente su propio juego y compartirlo con los estudiantes. Es decir, crea material de aprendizaje adaptativo dentro de los dispositivos móviles. Además, Memorato le permite al docente generar datos estadísticos sobre los usuarios respecto a sus juegos, y de esta manera es posible llevar un seguimiento sobre el desempeño de los estudiantes.

El juego del memorama o juego de memoria es un juego clásico que contiene un conjunto de pares de cartas con el lado de la información oculta; esta información, tradicionalmente, son dibujos animados de personajes. Es decir, cada carta tiene un lado (anverso) con información y otro lado (reverso) con textura/color (todas las cartas tienen exactamente el mismo reverso). Las cartas deben desordenarse sin que se pueda distinguir el contenido de cada una, y deben acomodarse en filas y columnas para que los jugadores alternen turnos para encontrar pares de cartas en ese conjunto desordenado.

Las reglas del tradicional juego de memoria son las siguientes: pueden jugar una o más personas (jugadores), quienes alternan turnos secuencial-

mente, en su turno cada jugador selecciona y voltea dos cartas, compara si son iguales, de ser así, ambas cartas se remueven del juego. Ese jugador suma puntos y vuelve a jugar repitiendo el proceso hasta que las cartas sean diferentes o se terminen las cartas disponibles. Si las cartas no corresponden (son distintas), ambas se vuelven a colocar en la misma posición de donde fueron volteadas con el lado reverso visible. El siguiente jugador toma su turno, debería haber visto las cartas previas destapadas por los jugadores anteriores y decidir cuáles destapar. El jugador que al finalizar el juego haya destapado más parejas de cartas iguales gana el juego. En caso de solo ser un jugador, se puede jugar contra reloj para terminar el juego en menor tiempo.

Como el nombre lo indica, el juego consiste en ejercitar la memoria dado que hay que recordar las posiciones de las cartas previas. En Memorato la idea es personalizar el juego y en lugar de tener dibujos en el anverso de las cartas, utilizar información de cualquier tema que el docente desee abordar. Cada par de cartas consiste en una pregunta-respuesta. Esto significa que el juego es personalizado por cada docente respecto a qué material habrá en las cartas. Con esto, si un estudiante abre una carta con la pregunta, debe saber cuál es la respuesta, de lo contrario no le servirá saber en qué posiciones hay cartas previamente descubiertas.

8.2 Arquitectura de Memorato

Para la propuesta se usó el desarrollo denominado “diseño centrado en el usuario” (*User-Centered Design*, UCD), es decir, el proceso de diseño estuvo orientado hacia usuarios finales. Esta filosofía de diseño sigue el objetivo de creación de productos que resuelven necesidades concretas de sus usuarios.

La figura 55 ilustra la estructura principal de Memorato. Como se puede observar, existen dos actores principales, el docente y los estudiantes, quienes interactúan con el sistema. El docente tiene acceso a un editor gráfico donde define la información del juego que va a crear, como el conjunto de preguntas-respuestas, el área académica a la que corresponde, el nivel de dificultad y la edad mínima para jugarlo. Toda esta información se almacena en el repositorio de juegos que forma parte de la aplicación. Una vez

guardado el juego, se genera un identificador (ID) del juego con el cual es posible publicarlo. El ID puede ser compartido con los estudiantes, quienes podrán usar el juego con la actividad generada.

El módulo de juegos recomendados se utiliza para que el sistema recomiende al estudiante algunos juegos existentes en el repositorio; por ahora el sistema solo se basa en el perfil del usuario seleccionando aquellas áreas en las que acaba de jugar o en la marcada en el buscador. El sistema, además, utiliza la edad del estudiante para buscar, filtrar y presentar los juegos recomendados en función de su nivel de complejidad. Las ocurrencias encontradas están rankeadas por la popularidad del juego.

Figura 55. Estructura del sistema Memorato

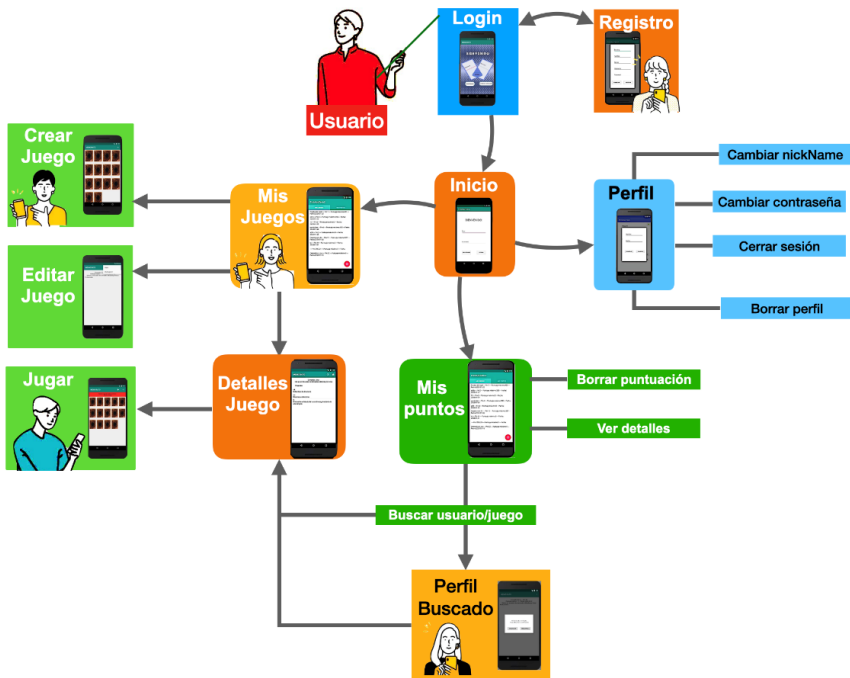


8.3 Diagrama de flujo de Memorato

La figura 56 muestra el diagrama de flujo de Memorato desde el punto de vista tanto del docente como del estudiante. Ambos usuarios deben registrarse inicialmente para hacer uso del sistema. El usuario debe autenticarse para ingresar. El usuario puede crear nuevos juegos y almacenarlos, para posteriormente editarlos (modificarlos) o incluso borrarlos. El sistema registra las actividades del usuario actualizando su perfil cuando accede a una acti-

vidad, y registra su tiempo de ejecución y puntuación obtenida. Cada actividad mantiene una lista de los mejores jugadores, tiempo y puntuaciones obtenidas.

Figura 56. Diagrama del flujo del Memorato



Las interfaces de usuario de la aplicación están organizadas por las categorías siguientes: entrada de datos, consulta de información y juego.

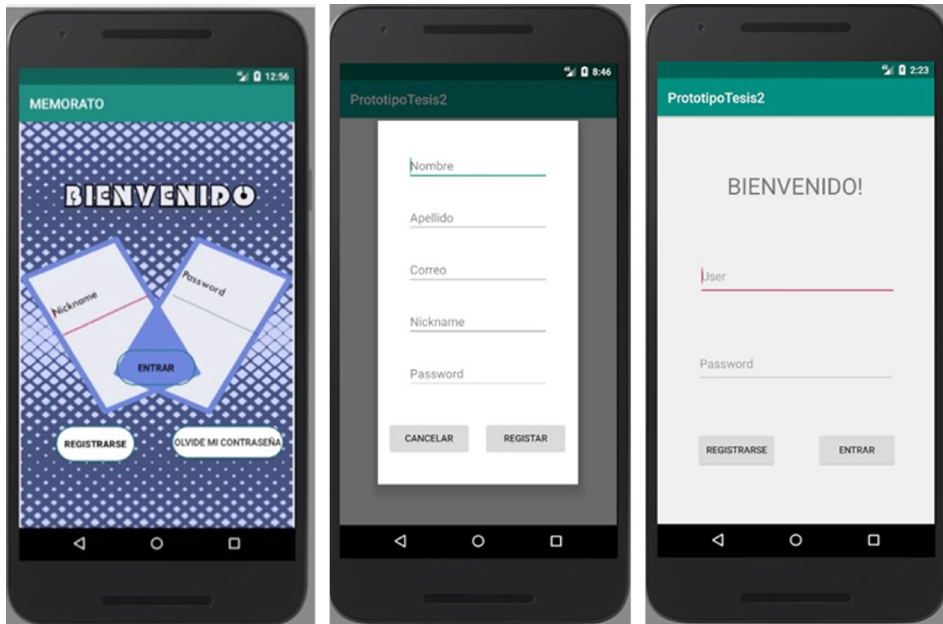
Interfaces para entrada de datos

Las interfaces para entrada de datos se agruparon en dos secciones: ingreso/registro y creación de juego, las cuales se describen a continuación.

1. Ingreso/registro. Como se mencionó anteriormente, el juego es personalizable, y para acceder al juego, el usuario debe estar registrado. La primera interfaz que el usuario visualiza es la pantalla de inicio (véase figura 57, izquierda); en esta interfaz se muestran tres opciones que sirven para entrar, registrarse

en el sistema y recordar la contraseña. La figura 57 (centro) muestra la interfaz para registrar a un usuario nuevo, y en la misma figura (derecha) se presenta la interfaz de autenticación, donde se solicita usuario y contraseña.

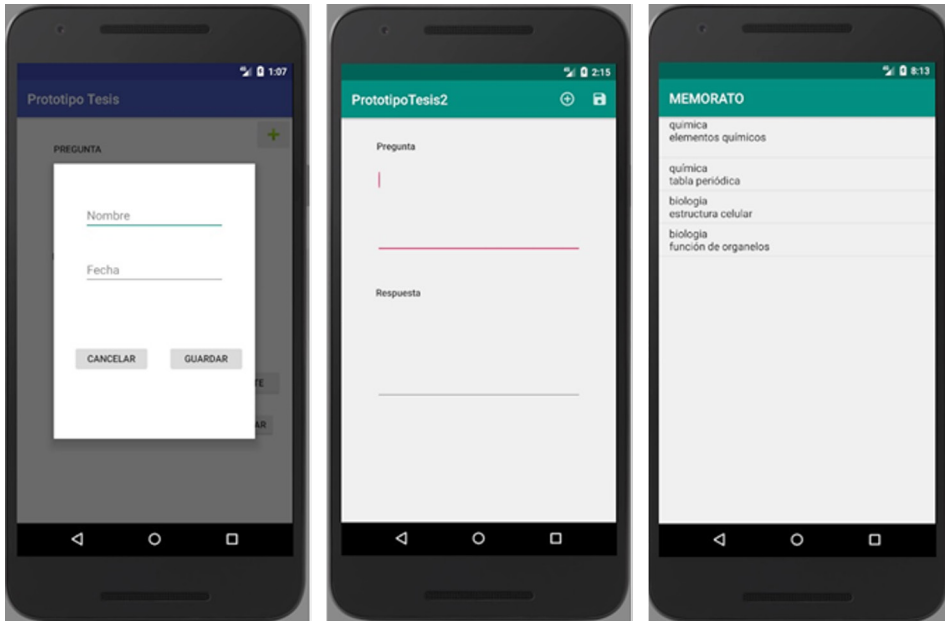
Figura 57. Interfaces para entrada de datos (ingreso / registro)



2. Creación de juego. Una vez autenticado, el usuario está en condiciones de crear un nuevo juego. La figura 58 muestra las tres interfaces que se utilizan para la creación de un juego:

- a) **Interfaz de captura de un juego** (lado izquierdo). Cada juego debe tener un título, además de haber seleccionado un área de las mostradas en el juego: matemáticas, español, inglés, ciencias naturales, etc.
- b) **Interfaz de captura del par pregunta-respuesta** (centro). El usuario selecciona un juego, y en esta interfaz puede introducir los pares de los que constará el juego; solo puede introducir un par a la vez, pregunta y respuesta.
- c) **Interfaz de lista de juegos con categorías** (lado derecho). El usuario puede visualizar los juegos y sus categorías.

Figura 58. Interfaces para entrada de datos (creación de juego)



Interfaces para consulta de información

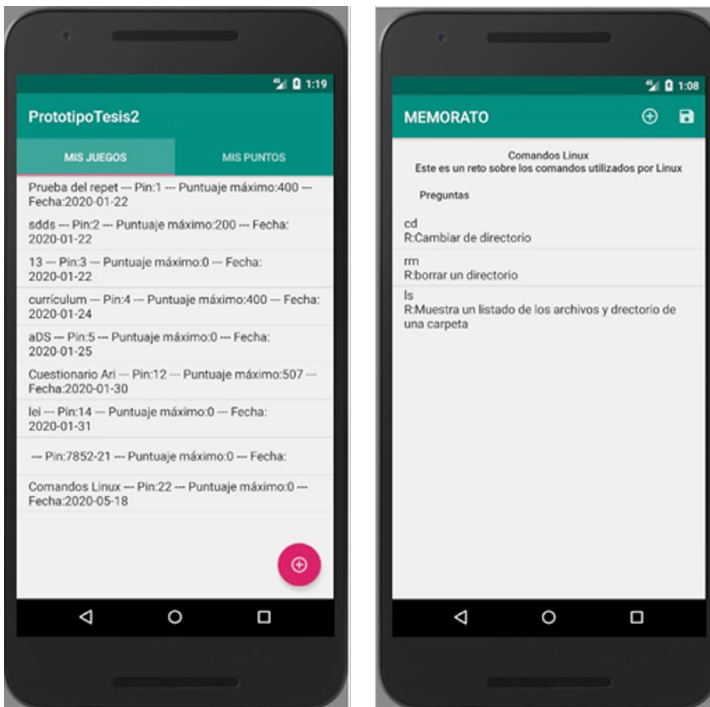
Estas interfaces representan el módulo principal de Memorato. Por cada usuario registrado, el sistema gestiona toda la información sobre los juegos creados, los juegos jugados, las estadísticas sobre los juegos diseñados; además es posible que el usuario pueda ver el detalle de cada juego capturado por él mismo (cada par es re-editable).

La información que se presenta al usuario sobre los juegos creados se hace a través de una interfaz, como se muestra en la figura 59. En esta figura (lado izquierdo) se presenta un listado de juegos creados por el usuario; en la parte superior hay dos pestañas llamadas: Mis juegos y Mis puntos. En Mis puntos, el usuario encontrará los juegos que él mismo ha jugado. Finalmente, en la esquina inferior derecha, el botón rojo que tiene un signo de + escrito en blanco, sirve para agregar un nuevo juego.

En esta misma figura 59 (lado derecho), se muestra el detalle de un juego creado con tres pares de pregunta-respuesta y la posibilidad de agregar nuevas preguntas. Esto puede conseguirse usando el botón con el signo

de + que se encuentra en la esquina superior derecha. Cada pregunta se describe en dos partes, la primera es la pregunta y la segunda —el renglón que comienza con R— es la respuesta. Finalmente, para regresar a las pantallas anteriores se usa el ícono de Guardar ubicado en la misma esquina superior derecha.

Figura 59. Interfaces para consulta de información

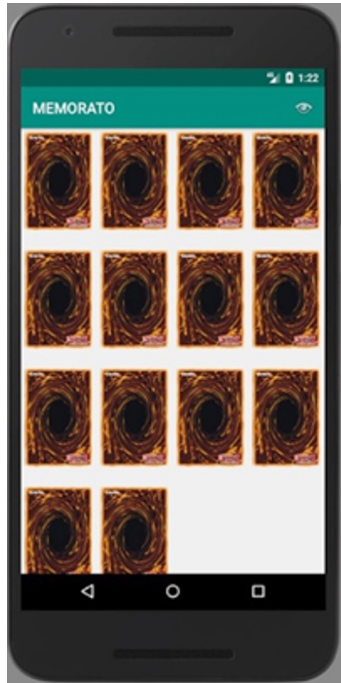


Interfaz del juego

La interfaz de un juego se muestra en la figura 60. Inicialmente todas las cartas están ocultas y bastará hacer clic sobre alguna de las cartas para que se muestre su contenido (anverso de la carta) en el centro de la pantalla, después de un breve instante, esta regresará a su posición inicial con su contenido aún visible. Cuando se destapa la segunda carta, sucede el mismo fenómeno y si las cartas corresponden, ambas desaparecerán, dejando un hueco en la posición que tenían. De lo contrario, ambas cartas volverán a

ocultarse (mostrando el reverso de las cartas) en la misma posición en la que se encontraba cada una.

Figura 60. Ejemplo de un juego de Memorato con siete pares de cartas



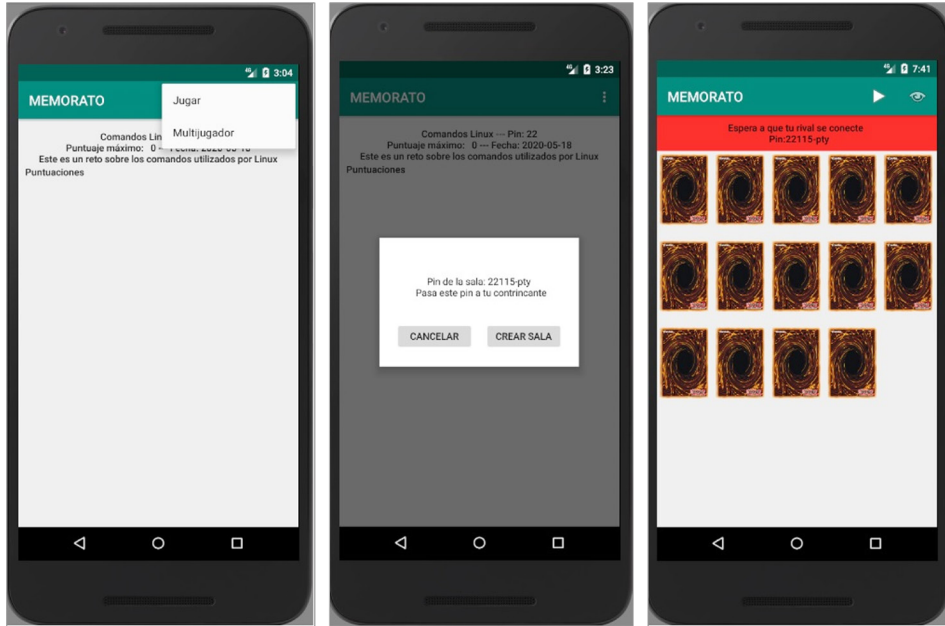
8.4 Versión multijugador

Memorato tiene opción de multijugador, es decir, es posible jugarlo entre dos personas de manera simultánea. Esto convierte al juego en más retador e interesante. Para habilitar esta opción, basta compartir el ID generado con el/la contrincante, como se puede apreciar en la figura 61.

La figura 61 (lado izquierdo) muestra las opciones de juego disponibles (Jugar o Multijugador); en el centro de la figura aparece la interfaz que muestra un ejemplo de los datos a compartir con un contrincante para jugar en versión multijugador. En el lado derecho, se muestra un juego en de-

sarrollo con un cintillo rojo en la parte superior, el cual indica que está esperando la conexión del segundo jugador.

Figura 61. Interfaces para jugar Memorato en modo multijugador



El desarrollo del juego es el siguiente, cada usuario tiene un turno intercalado, cuando el jugador en turno levanta una carta, el usuario contrinicante también observa la misma interfaz. Si pasado un tiempo el jugador perdiera la conexión, Memorato se dará cuenta y en ese momento sigue el desarrollo en versión de un solo jugador.

8.5 Sistema de recomendación

Memorato permite la búsqueda de juegos utilizando diferentes elementos: por el ID asignado al juego cuando fue creado, por palabras clave o por área del conocimiento. Para esto, se establecieron los siguientes requerimientos:

- Cada juego tiene un identificador único (ID) que el usuario creador del juego puede conocer en cualquier momento. Este debe ser compartido para poder comenzar en el juego correcto.
- La búsqueda por palabra clave revisa tanto el nombre del juego como las palabras en las preguntas-respuestas.
- Finalmente, por área del conocimiento permite buscar los juegos marcados con un área específica.

El resultado de la búsqueda es ordenado de acuerdo con la popularidad (número de veces que un juego ha sido jugado), por área del conocimiento principalmente jugada por ese usuario, y por número de preguntas en el juego (en caso de empate de las dos opciones previas).

8.6 Experimentación

Memorato fue probado con grupos de distintas áreas académicas: química, español y biología. Los participantes oscilaban entre los 18 y 20 años, y manifestaron emplear entre 8 y 12 horas al día en aplicaciones de celular (YouTube, redes sociales, etc.).

Para evaluar el impacto de esta aplicación, los estudiantes fueron examinados previamente, luego se les explicó el juego y se les dejó que jugaran. Finalmente, los estudiantes fueron examinados académicamente de nuevo. Los resultados obtenidos fueron: en cuestión del rendimiento académico, los estudiantes participantes mejoraron sus calificaciones previas; respecto a la aplicación, los estudiantes estuvieron entusiasmados con ella y más aún en el modo multijugador.

Referencias

- Aleven, V., & Koedinger, K. (2002). An effective metacognitive strategy: Learning by doing and explaining with a computer-based cognitive tutor. *Cognitive Science*, 26(2), 147-179.
- Ally, M. (2004). Foundations of educational theory for online learning. En T. Anderson (Ed.), *Theory and practice of online learning* (pp. 15-44). Athabasca University.
- Anderson, R., Boyle, C. F., Corbett, A. T., & Lewis, M. W. (1990). Cognitive modeling and intelligent tutoring. *Artificial Intelligence*, 42(1), 7-49. [https://doi.org/10.1016/0004-3702\(90\)90093-F](https://doi.org/10.1016/0004-3702(90)90093-F)
- A.P. Association. (1984). Diagnostic and statistical manual of mental disorders (4th ed.). American Psychiatric Association.
- APA. (2021). Emotion. *APA dictionary of Psychology*. <https://dictionary.apa.org/emotion>
- Arroyo, I., Woolf, B., Cooper, D., Burleson, W., Muldner, K., & Christopherson, R. (2009). Emotion sensors go to school. En Diminitrova, V., Mizoguchi, R., Du Boulay, B., & Graesser, A. (eds.), *Proceedings of the 14th International Conference on Artificial Intelligence in Education* (pp. 17-24). IOS Press.
- Bartlett, M. S., Littlewort, G., Frank, M., Lainscsek, C., Fasel, I., & Movellan, J. (2006). Fully automatic facial action recognition in spontaneous behavior. En *7th International Conference on Automatic Face and Gesture Recognition (FG06)* (pp. 223-230). IEEE.
- Black, M. (1937). Vagueness: An exercise in logical analysis. *Philosophy of Science*, 4(4), 427-455.
- Boden, M. A. (1977). *Artificial intelligence and natural man*. Basic Books.
- Bosch, J. (2000). *Design and use of software architectures: Adopting and evolving a product-line approach*. Pearson Education.
- Boulay, B. D. (2011). Towards a motivationally intelligent pedagogy: How should an intelligent tutor respond to the unmotivated or the demotivated? En *New perspectives on affect and learning technologies* (pp. 41-52). Springer.

- Brusilovsky, P., & Millán, E. (2007). User models for adaptive hypermedia and adaptive educational systems. En *The Adaptive web* (pp. 3–53) Springer-Verlag. https://doi.org/10.1007/978-3-540-72079-9_1
- Busso, C., Deng, Z., Yildirim, S., Bulut, M., Lee, C. M., Kazemzadeh, A., Lee, S., Neumann, U., & Narayanan, S. (2004). Analysis of emotion recognition using facial expressions, speech and multimodal information. En *ICMI '04 Proceedings of the 6th International Conference on Multimodal Interfaces* (pp. 205-211).
- Carbonell, J. R. (1970). AI in CAI: An artificial intelligence approach to computer-aided-instruction. *IEEE Transactions on Man-Machine System*, 11(4), 190-202.
- Chellappa, R., Wilson, C. L., & Sirohey, S. (1995). Human and machine recognition of faces: A survey. *Proceedings of the IEEE*, 83(5), 705-741.
- Chen, Y. S., Kao, T. C., Sheu, J. P., & Chiang, C. Y. (2002). A mobile scaffolding-aid-based bird-watching learning system. En *Proceedings. IEEE International Workshop on Wireless and Mobile Technologies in Education* (pp. 15-22). IEEE.
- Clancey, W. J. (1979). Transfer of rule-based expertise through a tutorial dialogue (Doctoral dissertation). *Computer Science Stanford, CA, Stanford University*. (No. STAN-CS-79-769).
- Clark, J., Yallop, C., & Fletcher, J. (2007). *An introduction to phonetics and phonology*. Wiley-Blackwell.
- Coffield, F., Moseley, D., Hall, E. & Ecclestone, K. (Ed.). (2004). *Learning styles and pedagogy in post-16 learning: A systematic and critical review*. Learning and Skills Research Centre.
- Cohn, J. F. (2006). Foundations of human computing: Facial expression and emotion. En *Proceedings of the 8th International Conference on Multimodal Interfaces* (pp. 233-238).
- Conati, C., & Maclare, H. (2004). Evaluating a probabilistic model of student affect. En *International Conference on Intelligent Tutoring Systems* (pp. 55-66). Springer.
- Copeland, B. J. (2022). Artificial intelligence. *Encyclopaedia britannica*.
- Delac, K., & Grgic, M. (2007). *Face recognition, I-Tech education and publishing*. Vienna, Austria.
- Deleuze, G., & Guattari, F. (1988). *A thousand plateaus: Capitalism and schizophrenia*. Bloomsbury Publishing.
- Dellaert, F., Polzin, T., & Waibel, A. (1996). Recognizing emotion in speech. En *Proceeding of Fourth International Conference on Spoken Language Processing. ICSLP'96* (Vol. 3, pp. 1970-1973). IEEE.
- D'Mello, S., Dowell, N., & Graesser, A. (2009). Cohesion relationships in tutorial dialogue as predictors of affective states. En *Artificial intelligence in education* (pp. 9-16). los Press.
- D'Mello, S. K., Lehman, B., & Graesser, A. (2011). A motivationally supportive affect-sensitive autotutor. En *New perspectives on affect and learning technologies* (pp. 113-126). Springer.
- D'Mello, S., Picard, R. W., & Graesser, A. (2007). Toward an affect-sensitive autotutor. *IEEE Intelligent Systems*, 22(4), 53-61.

- DOF (2011, 19 de agosto). *Acuerdo número 592 por el que se establece la Articulación de la Educación Básica*.
- Ekman, P. (1972). Universals and cultural differences in facial expressions of emotion. En J. Cole (ed.), *Nebraska Symposium on Motivation 1971*. University of Nebraska.
- Ekman, P. (1999). Facial expressions. *Handbook of Cognition and Emotion*, 16(301), e320.
- Ekman, P., & Friesen, W. V. (1978). *Facial action coding system: Investigator's guide*. Consulting Psychologists.
- Ekman, P., Friesen, W. V., & Hager, J. C. (2002). Facial action coding system, a human face. *ETC What is the ETC*.
- Essa, I. A., & Pentland, A. P. (1995). Facial expression recognition using a dynamic model and motion energy. En *Proceedings of IEEE International Conference on Computer Vision* (pp. 360-367). IEEE.
- European-Commission. (2001). *The eLearning Action Plan designing tomorrow's education*. Commission of the European Communities. <https://eur-lex.europa.eu/LexUriServ/LexUriServ.do?uri=COM:2001:0172:FIN:EN:PDF>
- Felder, R. M. (1996). Matters of style. *ASEE Prism*, 6(4), 18-23.
- Felder, R. M., & Silverman, L. K. (1988). Learning and teaching styles in engineering education. *Engineering Education*, 78(7), 674-681.
- Felder, R. M., & Soloman, B. A. (2004). *Index of learning styles questionnaire*. North Carolina State University. <http://www.engr.ncsu.edu/learningstyles/ilsweb.html>
- Felder, R. M., & Spurlin, J. (2005). Applications, reliability and validity of the index of learning styles. *International Journal of Engineering Education*, 21(1), 103-112.
- Forbes-Riley, K., & Litman, D. J. (2009). Adapting to student uncertainty improves tutoring dialogues. En *AIED* (pp. 33-40).
- Gardner, H. (2000) Can technology exploit our many ways of knowing? En D. T. Gordon (Ed.), *The digital classroom: How technology is changing the way we teach and learn* (pp. 32 – 35). Harvard Education Letter.
- Graesser, A. C., Lu, S., Jackson, G. T., Mitchell, H. H., Ventura, M., Olney, A., & Louwerse, M. M. (2004). AutoTutor: A tutor with dialogue in natural language. *Behavior Research Methods, Instruments, & Computers*, 36(2), 180-192.
- Hasan, M. A., Noor, N. F. M., Rahman, S. S. B. A., & Rahman, M. M. (2020). The transition from intelligent to affective tutoring system: A review and open issues. *IEEE Access*, 8, 204612-204638.
- Haykin, S. (1999). *Neural networks: A comprehensive foundation* (2nd ed). Prentice-Hall.
- Holland, J.H. (1975). *Adaptation in natural and artificial systems*. University of Michigan.
- Ji, Q., Lan, P., & Looney, C. (2006). A probabilistic framework for modeling and real-time monitoring human fatigue. *IEEE Transactions on Systems, Man, and Cybernetics-Part A: Systems and Humans*, 36(5), 862-875.
- Kaliouby, R. E., & Robinson, P. (2005). Real-time inference of complex mental states from facial expressions and head gestures. En *Real-time vision for human-computer interaction* (pp. 181-200). Springer.
- Kasabov, N. K. (1998). *Foundations of neural networks, fuzzy systems, and knowledge engineering*. The MIT Press.

- Keefe, J. W., & Thompson, S. D. (1987). *Learning Style: Theory and Practice*. National Association of Secondary School Principals.
- Keegan, D. (2000). *M-Learning. How far has it come?* (Propuesta de proyecto de Ericsson para la Comisión Europea).
- Khan, M. M., Ward, R. D., & Ingleby, M. (2009). Classifying pretended and evoked facial expressions of positive and negative affective states using infrared measurement of skin temperature. *ACM Transactions on Applied Perception (TAP)*, 6(1), 1-22.
- Kohonen, T. (1990). The self-organizing map. *Proceedings of the IEEE*, 78(9), 1464-1480.
- Kulik, C. L. C., & Kulik, J. A. (1991). Effectiveness of computer-based instruction: An updated analysis. *Computers in Human Behavior*, 7(1-2), 75-94.
- Langner, O., Dotsch, R., Bijlstra, G., Wigboldus, D. H., Hawk S. T., & Van Knippenberg, A. D. (2010). Presentation and validation of the Radboud Faces Database. *Cognition and Emotion*, 24(8), 1377-1388. <https://doi.org/10.1080/02699930903485076>
- Lee, C. M., Narayanan, S., & Pieraccini, R. (2001). Recognition of negative emotions from the speech signal. En *IEEE Workshop on Automatic Speech Recognition and Understanding 2001, ASRU'01* (pp. 240-243). IEEE.
- Lehner, F., & Nosekabel, H. (2002). The role of mobile devices in E-Learning first experiences with a wireless E-Learning environment. En *Proceedings. IEEE International Workshop on Wireless and Mobile Technologies in Education* (pp. 103-106). IEEE.
- Lienhart, R., & Maydt, J. (2002, septiembre). An extended set of haar-like features for rapid object detection. In *Proceedings. International Conference on Image Processing* (vol. 1, pp. 900-903). IEEE.
- Littlewort, G. C., Bartlett, M. S., & Lee, K. (2007, noviembre). Faces of pain: Automated measurement of spontaneous allfacial expressions of genuine and posed pain. En *Proceedings of the 9th International Conference on Multimodal Interfaces* (pp. 15-21).
- Litzinger, T., Lee, S. H., Wise, J., & Felder, R. (2005). *A study of the reliability and validity of the Felder Soloman Index of Learning Styles*. *Proceedings of the 2005 American Society for Engineering Education Annual Conference & Exposition*, <https://doi.org/10.18260/1-2--15321>
- Luckin, R., Holmes, W., Griffiths, M. y Forcier, L. B. (2016). *Intelligence unleashed: An argument for AI in education*. Pearson. <https://www.pearson.com/content/dam/one-dot-com/one-dot-com/global/Files/about-pearson/innovation/Intelligence-Unleashed-Publication.pdf>
- Lukasiewicz, J. (1930). Philosophische Bemerkungen zu mehrwertigen Systemen des Aussagenkalküls. *Comptes rendus des séances de la Société des Sciences et des Lettres de Varsovie*, 51-77.
- Marín I., Hierro, E. (2013). *Gamificación*. Empresa Activa.
- Mavrikis, & M., Holmes, W. (2019). Intelligent learning environments: Design, usage and analytics for future schools. En Yu, S., Niemi, H., Mason, J. (Eds.), *Shaping future schools with digital technology: Perspectives on rethinking and reforming education*. Springer. https://doi.org/10.1007/978-981-13-9439-3_4
- McKimm, J., Jollie, C., & Cantillon, P. (2003). Web based learning. *BMJ*, 326(7394), 870-873.

- Mills, D. W. (2002). *Applying what we know: Student learning styles*. <http://www.csrnet.org/csrnet/articles/student-learning-styles.html>.
- Molnar, A. R. (1990). Computers in education: A historical perspective of the unfinished task. *THE Journal (Technological Horizons in Education)*, 6(3), 80-83.
- Munakata, T. (2008). *Fundamentals of the new artificial intelligence: Neural, evolutionary, fuzzy and more*. Springer Science & Business Media.
- Myers-Briggs (2010). *MBTI Basics*. The Myers-Briggs Foundation. <https://www.myers-briggs.org/my-mbti-personality-type/mbti-basics/>
- Negnevitsky, M. (2005). *Artificial intelligence: A guide to intelligent systems* (2nd ed.). Addison-Wesley.
- Nichols, M. (2008). E-learning in context. *E-Primer Series*, 1, 1-28.
- Nkambou, R., Bourdeau, J., & Mizoguchi, R. (2010). Introduction: What are intelligent tutoring systems, and why this book?. En Nkambou, R., Bourdeau, J., Mizoguchi, R. (Eds.), *Advances in intelligent tutoring systems* (pp. 1-12). Springer. https://doi.org/10.1007/978-3-642-14363-2_1
- Nwana, H. S. (1990). Intelligent tutoring systems: An overview. *Artificial Intelligence Review*, 4(4), 251-277.
- Oblinger, D. G., & Hawkins, B. L. (2005). IT myths: the myth about e-Learning. *Educause Review*, 40(4), 14-15.
- OECD, PISA. (2018). Programa para la Evaluación Internacional de Alumnos de la OCDE PISA 2018: Resultados. https://www.oecd.org/pisa/publications/PISA2018_CN_MEX_Spanish.pdf
- O'Reilly, T. (2006). Qué es web 2.0: Patrones del diseño y modelos del negocio para la siguiente generación del software. <http://sociedaddelainformacion.telefonica.es/jsp/articulos/detalle.jsp?elem=2146>.
- Pantoja Ospina, M. A., Duque Salazar, L. I., & Correa Meneses, J. S. (2013). Modelos de estilos de aprendizaje: Una actualización para su revisión y análisis. *Revista Colombiana de Educación*, (64), 79-105. <https://www.redalyc.org/articulo.oa?id=413634076004>
- Papert, S. (1980). *Mindstorms: Children, computers and powerful ideas*. Basic Books.
- Parvez, S. M., & Blank, G. D. (2008). Individualizing tutoring with learning style based feedback. En *International Conference on Intelligent Tutoring Systems* (pp. 291-301). Springer.
- Pekrun, R., Goetz, T., Titz, W., & Perry, R. P. (2002). Academic emotions in students' self-regulated learning and achievement: A program of qualitative and quantitative research. *Educational Psychologist*, 37(2), 91-105.
- Petrovica, S., Anohina-Naumeca, A., & Ekenel, H. K. (2017). Emotion recognition in affective tutoring systems: Collection of ground-truth data. *Procedia Computer Science*, 104, 437-444. <https://doi.org/10.1016/j.procs.2017.01.157>.
- Picard, R. W. (1995). *Affective computing*. (MIT Technical Report #321).
- Plutchik, R. (1980). Measurement implications of a psychoevolutionary theory of emotions. En *Assessment and modification of emotional behavior* (pp. 47-69). Springer.
- Polson, M. C., & Richardson, J. J. (2013). *Foundations of intelligent tutoring systems*. Psychology Press.

- Ponce, J., & Karahoca, A. (2009). *State of the art in face recognition*. I-Tech Education and Publishing.
- RAE. (2011). Emoción. *Diccionario de la lengua española*. <https://dle.rae.es/emoci%C3%B3n>
- Rodríguez, A. G., & Caro, E. M. (2003). Estilos de aprendizaje y e-learning: Hacia un mayor rendimiento académico. *Revista de Educación a Distancia (RED)*, (7).
- Ruiz-del-Solar, J., & Navarrete, P. (2005). Eigenspace-based face recognition: A comparative study of different approaches. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, 35(3), 315-325.
- Salinas, J. (2005). La gestión de los entornos virtuales de formación. *Seminario Internacional: La calidad de la formación en red en el Espacio Europeo de Educación Superior*, 1-21.
- Saucedo, L., Figueroa, K., & Rivera-Loaiza, C. (2020, noviembre). A classic game in modern time inside educational context. *Avances en Interacción Humano-Computadora*, 5(1) 85-89. <http://aihc.amexihc.org/index.php/aihc/article/view/72>
- SCORM (The Sharable Content Object Reference Model). (2008). <http://www.adlnet.org>.
- SEP (2021). Principales cifras del sistema educativo nacional 2019-2020. https://www.planeacion.sep.gob.mx/Doc/estadistica_e_indicadores/principales_cifras/principales_cifras_2019_2020_bolsillo.pdf
- Sharples, M., Corlett, D., & Westmancott, O. (2002). The design and implementation of a mobile learning resource. *Personal and Ubiquitous Computing*, 6(3), 220-234.
- Silvern, L. C. (1983). Review of *Informational Technology and Its Impact on American Education*. *Educational Technology*, 23(6), 22-24. <http://www.jstor.org/stable/44425054>
- Stash, N. V., Cristea, A. I., & De Bra, P. M. (2004). Authoring of learning styles in adaptive hypermedia: Problems and solutions. En *Proceedings of the 13th International World Wide Web Conference on Alternate Track Papers & Posters* (pp. 114-123).
- Stathopoulou, I. O., & Tsihrintzis, G. A. (2010). *Visual affect recognition* (vol. 214). IOS Press.
- Tosa, N., & Nakatsu, R. (1996). Life-like communication agent-emotion sensing character "MIC" and feeling session character "MUSE". En *Proceedings of the third IEEE International Conference on Multimedia Computing and Systems* (pp. 12-19). IEEE.
- Valstar, M., & Pantic, M. (2006). Fully automatic facial action unit detection and temporal analysis. En *2006 Conference on Computer Vision and Pattern Recognition Workshop (CVPRW'06)* (pp. 149-149). IEEE.
- VanLehn, K. (1988). Student modeling. En Polson, M.C., Richardson, J. J. (Eds.), *Foundations of Intelligent Tutoring Systems*, 55, 78.
- Viola, P., & Jones, M. (2001). Rapid object detection using a boosted cascade of simple features. En *Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. CVPR 2001* (vol. 1, pp. 511-518). IEEE.
- Woolf, B. P. (2009). *Building intelligent interactive tutors: Student-centered strategies for revolutionizing e-learning*. Morgan Kaufmann.

- Woolf, B., Burelson, W., & Arroyo, I. (2007). Emotional intelligence for computer tutors. En *Workshop on modeling and scaffolding affective experiences to impact learning at 13th international conference on artificial intelligence in education*, Los Angeles, California, pp. 6-15.
- Wundt, W. (2008). *An introduction to psychology*. Muller Press. (trabajo original publicado en 1912).
- Yacoob, Y., & Davis, L. S. (1996). Recognizing human facial expressions from long image sequences using optical flow. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 18(6), 636-642.
- Zadeh, L. (1965). Fuzzy sets. *Information and Control*, 8(3), 338-353.
- Zatarain Cabada, R., & Barrón Estrada, M. L. (2011). Herramienta de autor para la identificación de estilos de aprendizaje utilizando mapas auto-organizados en dispositivos móviles. *Revista Electrónica de Investigación Educativa*, 13(1), 43-55.
- Zatarain-Cabada, R., Barrón-Estrada, M. L., Angulo, V. P., García, A. J., & García, C. A. R. (2010). A learning social network with recognition of learning styles using neural networks. En *Mexican Conference on Pattern Recognition* (pp. 199-209). Springer.
- Zatarain-Cabada, R., Barron-Estrada, M. L., Zatarain-Cabada, R., & García, C. A. R. (2009). A fuzzy-neural network for classifying learning styles in a web 2.0 and mobile learning environment. En *2009 Latin American Web Congress* (pp. 177-182). IEEE.
- Zatarain Cabada, R., Barrón Estrada, M. L., & Reyes García, C. A. (2011). EDUCA: A web 2.0 authoring tool for developing adaptive and intelligent tutoring systems using a Kohonen network. *Expert Systems with Applications*, 38(8), 9522-9529.
- Zeng, Z., Pantic, M., Roisman, G. I., & Huang, T. S. (2008). A survey of affect recognition methods: Audio, visual, and spontaneous expressions. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 31(1), 39-58. <https://doi.org/10.1109/TPAMI.2008.52>
- Zhao, W., Chellappa, R., Phillips, P. J., & Rosenfeld, A. (2003). Face recognition: A literature survey. *ACM Computing Surveys (CSUR)*, 35(4), 399-458.
- Zywno, M. (2003). A contribution to validation of score meaning for Felder Soloman's Index of Learning Styles. En *Proceedings of the 2003 ASEE Annual Conference* (pp. 8-31).

Sobre los autores

RAMÓN ZATARAIN CABADA

Doctor en Ciencias de la Computación y profesor-investigador en el Tecnológico Nacional de México-Instituto Tecnológico de Culiacán. Es investigador nacional nivel 1 del Sistema Nacional de Investigadores (SNI) de Conacyt y miembro de la Academia Mexicana de Computación y de la Red Temática de Inteligencia Computacional Aplicada. Línea de investigación: sistemas inteligentes.

ORCID: <https://orcid.org/0000-0002-4524-3511>

MARÍA LUCÍA BARRÓN ESTRADA

Doctora en Ciencias de la Computación y profesora-investigadora en el Tecnológico Nacional de México - Instituto Tecnológico de Culiacán. Es investigadora nacional nivel 3 del Sistema Nacional de Investigadores (SNI) de Conacyt y miembro de la Academia Mexicana de Computación, de la Sociedad Mexicana de Ciencia de la Computación, A. C. y de la Red Temática de Inteligencia Computacional Aplicada. Línea de investigación: tecnologías computacionales inteligentes aplicadas a la educación.

ORCID: <https://orcid.org/0000-0002-3856-9361>

KARINA MARIELA FIGUEROA MORA

Doctora en Ciencias de la Computación y profesora-investigadora en la Facultad de Ciencias Físico-Matemáticas de la Universidad Michoacana de

San Nicolás de Hidalgo. Es investigadora nacional nivel 1 del Sistema Nacional de Investigadores (SNI) de Conacyt, miembro de la Academia Mexicana de Computación, de la Red Temática de Inteligencia Computacional Aplicada y de la Sociedad Mexicana de Ciencia de la Computación, A. C. ORCID: <https://orcid.org/0000-0002-4680-5950>

Índice de figuras

Figura 1. Rango de los valores en <i>a</i>) un sistema booleano (0 y 1) y <i>b</i>) un sistema difuso con un número infinito de valores	21
Figura 2. Representación de conjuntos difusos para valores de temperatura Baja, Normal y Alta	22
Figura 3. Diferentes cercos (<i>hedges</i>) para un conjunto difuso como temperatura normal	23
Figura 4. Sistema difuso para obtener la propina en un restaurante .	24
Figura 5. Diagrama de una neurona	26
Figura 6. Red neuronal artificial <i>feedforward</i>	27
Figura 7. Datos de entrada y salida deseada de la red neuronal . . .	29
Figura 8. Red neuronal artificial Perceptron para operación lógica OR	30
Figura 9. Selección por técnica de ruleta	33
Figura 10. Operaciones de cruzamiento (arriba) y mutación (abajo)	33
Figura 11. Tres tipos de Aprendizaje no presencial	38
Figura 12. Áreas de conocimiento relacionadas con los sistemas tutores inteligentes	47
Figura 13. Arquitectura típica de un sistema tutor inteligente	47
Figura 14. Estructura de un ambiente adaptable de aprendizaje . . .	50
Figura 15. Rueda de emociones	64
Figura 16. Imágenes visibles e infrarrojas	67
Figura 17. FTFP sobre cara humana y sobre mapa de los músculos faciales humanos	67

Figura 18. Las cuatro interfaces de entrada	71
Figura 19. Arquitectura principal de EDUCA	77
Figura 20. Interfaz de bienvenida al usuario	79
Figura 21. Estructura de un curso creado en EDUCA	80
Figura 22. Edición de un tema con recomendaciones	81
Figura 23. Creación de cuestionarios en EDUCA	82
Figura 24. Especificación de variable lingüística	83
Figura 25. Identificación de estilos de aprendizaje.	84
Figura 26. Visualización en simulador de un curso de matemáticas	85
Figura 27. Despliegado de curso en dispositivo móvil (simulación) .	86
Figura 28. Despliegado de diferentes componentes	86
Figura 29. Red neuronal durante el entrenamiento (datos hipotéticos)	90
Figura 30. Arquitectura en capas de la plataforma Zamná	99
Figura 31. Diagrama de despliegue	101
Figura 32. Estructura de Zamná	103
Figura 33. Interfaces de Zamná	104
Figura 34. Comunidad de Programación de Sistemas.	105
Figura 35. Proceso de creación de un curso	107
Figura 36. Interfaz del editor de cursos	108
Figura 37. Horas lectivas estudiantes tercer año de primaria	113
Figura 38. Porcentaje de resultados 2006-2011	113
Figura 39. Tendencias en el desempeño en lectura, matemáticas y ciencias	114
Figura 40. Esquema de los elementos de Fermat	118
Figura 41. Arquitectura de software para Fermat	119
Figura 42. Estructura del STI de Fermat	120
Figura 43. Mecanismo de evaluación del tutor	126
Figura 44. Interfaz del tutor inteligente Fermat	127
Figura 45. Las tres variables de entrada del sistema difuso	128
Figura 46. Conjuntos difusos de variables Error, Ayuda y Tiempo .	129
Figura 47. Sistema reconocedor de emociones	131
Figura 48. Rostro con las 22 distancias usadas para reconocer emociones	133

Figura 49. Archivo con emociones y distancias para entrenar a la red neuronal	133
Figura 50. Ejemplo de imágenes de la base de datos Grimace	134
Figura 51. Ejemplo de un modelo mostrando cinco emociones en JAFFE.	135
Figura 52. Ejemplo de un modelo mostrando ocho emociones en RaFD	136
Figura 53. Conversión y normalización de las distancias.	136
Figura 54. Conversión de las distancias normalizadas a pesos ya entrenados	137
Figura 55. Estructura del sistema Memorato	142
Figura 56. Diagrama del flujo del Memorato	143
Figura 57. Interfaces para entrada de datos (ingreso / registro)	144
Figura 58. Interfaces para entrada de datos (creación de juego)	145
Figura 59. Interfaces para consulta de información	146
Figura 60. Ejemplo de un juego de Memorato con siete pares de cartas	147
Figura 61. Interfaces para jugar Memorato en modo multijugador	148

Índice de tablas

Tabla 1. Modelos de estilos de aprendizaje	59
Tabla 2. Clasificación de modelos de estilos de aprendizaje	59
Tabla 3. Comparación de aplicaciones web 1.0 y su similar en la web 2.0	95
Tabla 4. Requisitos de calidad de Zamná	97
Tabla 5. Casos de uso de Zamná	98
Tabla 6. Requisitos funcionales de la red social de Fermat	117
Tabla 7. Requisitos del sistema tutor inteligente	117
Tabla 8. Mapeo de casos de uso de Fermat a requisitos funcionales .	117
Tabla 9. Valores difusos para variable Dificultad	129

Sistemas inteligentes aplicados a la educación, Ramón Zatarain Cabada, María Lucía Barrón Estrada, Karina Mariela Figueroa Mora, publicado por Ediciones Comunicación Científica, S. A. de C. V., se terminó de imprimir en marzo de 2023, en los talleres de Ultradigital Press, S. A. de C. V., Centeno 195, Col. Valle del Sur, 09819, Ciudad de México. El tiraje fue de 500 ejemplares impresos y en versión digital en los formatos PDF, EPUB y HTML.

Las computadoras se han utilizado en la educación desde el siglo pasado con el fin de proporcionar a los estudiantes herramientas que contribuyan en su proceso de aprendizaje. El desarrollo de sistemas computacionales para apoyar el proceso educativo involucra diferentes áreas de conocimiento para generar recursos educativos digitales adaptables a las necesidades personales de los estudiantes. Estos sistemas educativos han evolucionado incorporando diferentes elementos acordes al desarrollo tecnológico y es común encontrar sistemas que utilizan elementos multimedia, realidad aumentada, ambientes virtuales, y sistemas de recomendación, entre otros, para ofrecer experiencias únicas a los estudiantes.

En las últimas décadas, los avances de la inteligencia artificial, la disponibilidad de dispositivos móviles y la popularidad de internet han propiciado el desarrollo de sistemas educativos con características únicas que proveen instrucción personalizada de acuerdo con las necesidades del usuario, así como su acceso en cualquier momento y lugar.

En esta obra se presentan conceptos y temas básicos de algunas áreas de conocimiento necesarias para desarrollar sistemas computacionales inteligentes aplicados a la educación, así como su implementación en cuatro trabajos de investigación que incorporan diferentes técnicas y módulos inteligentes (lógica difusa, algoritmos genéticos, o redes neuronales) con el fin de proporcionar al usuario una experiencia de aprendizaje personalizada más adecuada a sus necesidades cognitivas y afectivas.



Ramón Zatarain Cabada es Doctor en Ciencias de la Computación, profesor investigador en la División de Estudios de Posgrado e Investigación en el Instituto Tecnológico de Culiacán del Tecnológico Nacional de México, miembro del Sistema Nacional de Investigadores (SNI) del Conacyt y de la Academia Mexicana de Computación.



María Lucía Barrón Estrada es Doctora en Ciencias de la Computación, profesora investigadora en la División de Estudios de Posgrado e Investigación en el Instituto Tecnológico de Culiacán del Tecnológico Nacional de México, miembro del Sistema Nacional de Investigadores (SNI) del Conacyt, la Academia Mexicana de Computación y la Sociedad Mexicana de Ciencia de la Computación.



Karina Mariela Figueroa Mora Doctora en Ciencias de la Computación, profesora investigadora en Facultad de Ciencias Físico Matemáticas de la Universidad Michoacana de San Nicolás de Hidalgo, miembro del Sistema Nacional de Investigadores (SNI) del Conacyt, la Academia Mexicana de Computación y la Sociedad Mexicana de Ciencia de la Computación.



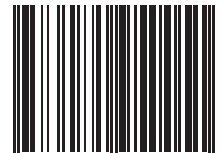
Dimensions



DOI.ORG/10.52501/CC.096



ISBN-13: 978-607-59500-5-1



9 786075 950051



**COMUNICACIÓN
CIENTÍFICA**

PUBLICACIONES
ARBITRADAS
HUMANIDADES, SOCIALES Y CIENCIAS

www.comunicacion-cientifica.com