

Centro Universitario "José Martí Pérez" Sancti-Spíritus
Facultad de Ingeniería



Maestría en Nuevas Tecnologías para la Educación

Título: Conjunto de agentes inteligentes para
proporcionar adaptabilidad a APA-Prolog

Autor:

Lic. Ángel Aljadis Díaz Piña

Tutor:

Dra. Lidia Rosa Ríos Rodríguez

Sancti-Spíritus, Mayo 2010
"Año 52 de la Revolución"

PENSAMIENTO

“En nuestros países ha de hacerse una revolución radical en la educación”.

José Martí.

DEDICATORIA

A mis padres, por su gran sacrificio y apoyo.

A mi familia.

A mis amigos.

A GRADDECIMIENTOS

A mi tutora Dra. Lidia Rosa Ríos Rodríguez, por su apoyo.

A Yunalis por su gran ayuda.

A Yandira por su ayuda.

RESUMEN

RESUMEN

Uno de los retos actuales de la educación en nuestro país, consiste no solo en hacer más eficiente el aprendizaje sino también en obtener egresados capaces de organizar su propio aprendizaje e interactuar con la información y el conocimiento de forma exitosa. Esto ha provocado cambios en los instrumentos de transmisión del conocimiento. Entre los instrumentos para el aprendizaje visual se encuentran los mapas conceptuales, estos no solo transmiten información básica sino que permiten ver las relaciones, estructuras, modelos y características. La programación declarativa es un paradigma de programación con características propias y cuando se enseña después de haber trabajado por mucho tiempo con otro estilo se hace muy difícil su comprensión. Se cuenta con un Sistema de autoaprendizaje para la Programación Lógica (APA-ProLog) diseñado e implementado por el grupo de desarrollo de software para la educación del CUSS, el cual no es capaz de adaptarse a las necesidades de los educandos. Por tanto se hace necesario diseñar e implementar un conjunto de agentes inteligentes para proporcionar adaptabilidad a APA-Prolog.

ABSTRACT

ABSTRACT

In our country, one of the current challenges of education consists not only in making the learning process more efficient but also in providing graduate professionals able to arrange their own learning style and to interact with information and knowledge in a successful way. This has caused changes in the instruments of knowledge transmission. Among the instruments for the visual learning are the conceptual maps; these not only transmit basic information but rather allow to see the relationships, structures, models and features. The declarative programming is a programming paradigm with its own features and when it is taught after having worked for a long time with another style, it becomes very difficult to understand. There is a self learning System for the Logical Programming (APA-ProLog), designed and implemented by the CUSS Software Development Group for education, which is not able to adapt to the necessities of the students. Therefore it becomes necessary to design and to implement a group of intelligent agents to provide adaptability to APA-Prolog.

ÍNDICE

| | |
|---|-----------|
| ÍNDICE | 1 |
| INTRODUCCIÓN | 1 |
| CAPITULO 1. SISTEMAS INTELIGENTES EN LA EDUCACIÓN..... | 7 |
| 1.1 UTILIZACIÓN DE LAS COMPUTADORAS EN LA EDUCACIÓN..... | 7 |
| 1.1.1 <i>Enfoque de la Informática como medio de enseñanza.....</i> | 10 |
| 1.2 LOS AGENTES INTELIGENTES | 11 |
| 1.2.1 <i>Los Agentes Inteligentes en ambientes de enseñanza aprendizaje</i> | 14 |
| 1.3 SISTEMAS ADAPTATIVOS | 17 |
| 1.3.1 <i>Elementos a tener en cuenta para lograr la adaptación.....</i> | 21 |
| 1.3.2 <i>Sistemas Adaptativos en la Enseñanza.....</i> | 23 |
| 1.4 FUNDAMENTACIÓN DE LA PROPUESTA DE DISEÑO DEL ENTORNO DE APRENDIZAJE | 24 |
| 1.4.1 <i>Enfoque histórico-cultural.....</i> | 25 |
| 1.5 SELECCIÓN DE LA TECNOLOGÍA ADECUADA | 27 |
| 1.5.1 <i>Las Tecnologías Web.....</i> | 29 |
| 1.5.2 <i>ZOPE como servidor de aplicaciones Web.....</i> | 30 |
| 1.5.3 <i>MySQL como servidor de bases de datos</i> | 32 |
| 1.5.4 <i>Ventajas de MySQL.....</i> | 32 |
| 1.5.5 <i>Implementación de los agentes inteligentes.....</i> | 33 |
| 1.6 CONCEPCIÓN GENERAL DE APA-PROLOG | 34 |
| 1.7 CONCLUSIONES PARCIALES | 36 |
| CAPITULO 2. DISEÑO E IMPLEMENTACIÓN DEL SISTEMA Y LOS AGENTES INTELIGENTES PARA APA-PROLOG | 37 |
| 2.1 DESCRIPCIÓN DEL SISTEMA PROPUESTO | 37 |
| 2.1.1 <i>Actores del sistema.....</i> | 38 |
| 2.1.2 <i>Diagrama de casos de usos del sistema.....</i> | 38 |
| 2.1.3 <i>Descripción de los casos de usos del sistema</i> | 41 |
| 2.1.4 <i>Diseño de la base de datos</i> | 44 |
| 2.1.5 <i>Diagramas de navegación</i> | 45 |
| 2.2 ARQUITECTURA BÁSICA DE UN AGENTE | 47 |
| 2.2.1 <i>Agente Adis.....</i> | 47 |
| 2.2.2 <i>Agente Tivo.....</i> | 51 |
| 2.2.3 <i>Agente Teo.....</i> | 52 |
| 2.2.4 <i>Agente Tica.....</i> | 53 |
| 2.2.5 <i>Agente Eva.....</i> | 57 |
| 2.2.5.1 <i>Evaluadores</i> | 57 |
| 2.3 MODELO PARA LA SOLUCIÓN COOPERATIVA | 58 |
| 2.4 ANÁLISIS DE LOS RESULTADOS VALORADOS POR EXPERTOS | 62 |
| 2.4.1 <i>Procesamientos estadísticos utilizados.....</i> | 62 |
| 2.4.2 <i>Valoraciones de los expertos</i> | 62 |
| 2.5 CONCLUSIONES PARCIALES | 66 |
| CONCLUSIONES..... | 67 |
| RECOMENDACIONES | 68 |
| REFERENCIAS BIBLIOGRÁFICAS..... | 69 |
| GLOSARIO | 77 |
| ANEXOS..... | 81 |

INTRODUCCIÓN

Las actuales generaciones transitan una época matizada por la información donde se habla comúnmente, en escenarios educativos diversos, del desarrollo de las potencialidades del ser humano, de la autonomía del alumno y de la necesidad de aprender a aprender [LOP07]. El fácil acceso a la información y su distribución por medios electrónicos multiplica el impacto formativo de las instituciones de educación.

La mayor interacción entre las comunidades académicas permite un proceso continuo de mejoramiento de la calidad educativa; la apertura a la interacción mundial potencia los procesos de transformación de las instituciones educativas [ALM07]. Es por ello que hacer de la virtualización un instrumento al servicio de la Nueva Universidad es uno de los propósitos de la dirección de Informatización del Ministerio de Educación Superior en Cuba.

En este contexto no debe perderse de vista, que diseñar recursos para la enseñanza bajo un esquema de “talla única” conllevaría a proveer a los estudiantes de recursos didácticos no necesariamente acordes a sus características, conocimientos previos o nivel de desarrollo cognitivo, en otras palabras a sus necesidades individuales de formación [BER04]. Otro elemento a tomar en cuenta es la forma en que se organiza el contenido. Muchas veces los estudiantes o cualquier persona que desea investigar se sienten defraudados cuando navegan por la Web y se declaran perdidos dentro de la tela de araña en que se transforma la información cuando redundan los hipervínculos y no existe una manera clara de navegar.

Los Sistemas Hipermedia Adaptativos (SHA) y las Web adaptativas representan un área de interés creciente de investigación en las Tecnologías de la Información y las Comunicaciones (TIC). Los investigadores de los SHA se enfocan al estudio de teorías, técnicas e innovaciones tecnológicas que permitan la personalización y adaptación de la información a las necesidades específicas de cada usuario [GRI08]. Diversas técnicas de Inteligencia Artificial (IA), entre las que descuellan los agentes inteligentes, se emplean con estos fines.

La palabra “agente” se escucha con frecuencia en las comunidades actuales dedicadas a las ciencias de la computación y al desarrollo de aplicaciones. En este caso se refiere a agentes de software, los que simulan a los agentes humanos y son el resultado de investigaciones en el campo de la ingeniería de software y la inteligencia artificial.

Del mismo modo que un agente humano tiene ojos, oídos y otros órganos para percibir el medio y manos, boca, piernas y otras partes del cuerpo para actuar sobre él, un agente de software percibe el medio a través de sensores y actúa sobre ese medio a través de acciones [RUS95] con la diferencia de que en este caso la percepción y la acción se realizan a través de segmentos de código. En otras palabras, son entidades computacionales que existen físicamente en forma de programas que se ejecutan sobre computadoras [WEI99].

Técnicamente, los agentes de software exhiben muchos beneficios, entre ellos:

- Modularidad. (Permitiendo adicionar y eliminar elementos sin afectar otros)
- Velocidad. (La ejecución concurrente de programas incrementa la velocidad de ejecución de todo el sistema)
- Otras, tales como mantenibilidad, reusabilidad e independencia de la plataforma [PEÑ04b].

Por otro lado, el reto actual de la educación en nuestro país consiste no solo en hacer más eficiente el aprendizaje sino también en obtener egresados capaces de organizar su propio aprendizaje e interactuar con la información y el conocimiento de forma exitosa. Esto ha provocado cambios en los instrumentos de transmisión del conocimiento y la interpretación de las representaciones visuales es una necesidad de las actuales y futuras generaciones.

En la Universidad Central de Las Villas, desde 1976 se iniciaron trabajos destinados a desarrollar materiales de Educación Asistida por Computadora (EAC). Un primer resultado fue el sistema CED (Cibernética Electrónica Didáctica) para la minicomputadora CID-300, que vio la luz en 1981, con una versión posterior para microcomputadoras (Micro-CED) [VAL87]. La primera versión sirvió más bien como

medio de desarrollo y experimentación, debido a las limitaciones naturales que supone la enseñanza basada en una minicomputadora poco disponible, lo que enfatiza el hecho de que el desarrollo de la EAC adquiere un carácter verdaderamente relevante al surgir las microcomputadoras.

Los estudios sobre el tema continuaron y en 1981 se concluyen los siguientes productos:

- SESE, un sistema de apoyo a la enseñanza y el aprendizaje de sistemas expertos, el cual está compuesto por 3 módulos:
 - TeachShell. Una máquina de inferencia.
 - UCShell. Un medio integrado para el desarrollo de Sistemas Expertos.
 - Compiler. Un compilador de líneas que genera código para TeachShell y UCShell. (Artículo de Mateo.)
- SEP, un sistema para la enseñanza del paradigma de la Programación Lógica.
- Progen, un generador de los módulos que componen un sistema experto escrito en lenguaje Prolog, que se rige por los principios enunciados en la metodología elaborada por el autor.

La metodología, unida al sistema Progen y la componente de SESE denominada UCShell, aparte de sus valores didácticos, presentan un importante valor de uso, desde el punto de vista de la producción de software en general.

Años más tarde, y como resultado del esfuerzo del grupo RedWay se obtuvo el Sistema de Enseñanza Personalizada A Distancia (SEPAD), importante herramienta que se ha utilizado en varios centros de educación superior del país.

Actualmente también se utiliza con fines docentes software elaborado por los propios estudiantes ya sea a través de trabajos de curso o trabajos de diploma. En la asignatura Prolog específicamente los ejemplos más importantes son:

- Entrenador inteligente sobre límite de funciones. LIM V1.0, resultado de un trabajo de diploma en el curso 2002-2003.
- Sistema para la estimación de fallas en Sistema Eléctricos de Potencia, producto de una tesis de maestría.
- Número y numerales, se obtuvo a partir del examen de premio de la asignatura Programación Lógica en el año 2002.

Desde el curso 2008-2009 se cuenta con otra herramienta para el Autoaprendizaje de la Programación Lógica, confeccionada por el grupo de desarrollo de software del CUSS, denominada APA-ProLog, la que persigue siguientes objetivos:

1. Aprovechar las ventajas que brindan las computadoras y las TICs para crear ambientes virtuales de enseñanza aprendizaje sin restricciones de espacio y tiempo.
2. Lograr un proceso de enseñanza-aprendizaje activo y significativo, protagonizado por el estudiante, facilitando los recursos que les permitan convertir la información en conocimiento.
3. Obtener un ambiente que tome en cuenta los conocimientos previos del estudiante para brindarle una atención personalizada.
4. Apoyar el proceso de enseñanza-aprendizaje de la programación Lógica.

Este ambiente utiliza los mapas conceptuales como forma de organizar el contenido que se quiere hacer llegar a los estudiantes e incluye diversos recursos informáticos (entrenadores, tutoriales, simuladores, presentaciones, un espacio para el intercambio y otros), sin embargo no toma en cuenta los conocimientos previos de los estudiantes, es decir la información llega a todos por igual.

Los sistemas adaptativos se distinguen por la posibilidad de individualizar el acceso a la información de manera que esta se acomode y responda a la diversidad y necesidades

de los usuarios, o sea, es la información la que se ajusta a las necesidades del alumno y no al revés como suele ocurrir.

Esta situación nos lleva al **siguiente problema de investigación**:

¿Cómo proporcionar adaptabilidad al Sistema de Autoaprendizaje para la Programación Lógica (APA-Prolog) de modo que permita personalizar la atención al estudiante?

Para darle solución al problema planteado se formula la siguiente **hipótesis**:

Si se implementa un conjunto de agentes inteligentes para proporcionar adaptabilidad al ambiente de autoaprendizaje asistido por computadora se contribuye a personalizar la atención al estudiante.

Se toma como objeto de estudio el proceso de autoaprendizaje y como campo de acción el proceso de autoaprendizaje asistido por computadora.

Para corroborar dicha hipótesis se estableció como **objetivo general**:

Implementar un conjunto de agentes inteligentes que proporcionen adaptabilidad a APA-Prolog permitiendo personalizar la atención al estudiante.

.

Para cumplir el objetivo trazado se proponen los siguientes objetivos específicos:

- ❖ Determinar un marco teórico acerca de los antecedentes sobre los agentes inteligentes y técnicas para lograr una atención personalizada al estudiante.
- ❖ Implementar un conjunto de agentes inteligentes que permitan una atención personalizada al estudiante en la Programación Lógica.
- ❖ Validar a través del criterio de experto el conjunto de agentes inteligentes.

Métodos de Investigación:

- Análisis histórico lógico: para la determinación y ordenación de antecedentes relacionados con la historia del problema de investigación.
- Inducción deducción: para determinar regularidades y sacar inferencias sobre el objeto de estudio.
- Estructural sistémico: para determinar que metodología se necesita emplear en la concepción del trabajo.
- Modelación: para diseñar el software didáctico.
- Análisis documental: se utilizó para analizar los documentos utilizados.
- Prueba pedagógica: dirigida a los alumnos para determinar las principales insuficiencias del aprendizaje de los conocimientos relacionados con la programación lógica.
- Consulta a expertos: para la evaluación del producto informático elaborado.

Población y muestra:

Estudiantes de 4to año de la carrera Ingeniería Informática del CUSS.

Estructura del cuerpo de la Tesis:

- Capítulo I: "Sistemas Inteligentes en la Educación"
- Capítulo II: "Diseño e implementación del Sistema y los Agentes Inteligentes para APA-Prolog".

CAPITULO 1. SISTEMAS INTELIGENTES EN LA EDUCACIÓN

1.1 Utilización de las computadoras en la educación

En los primeros sistemas de enseñanza asistida por computadora, conocidos como lineales, el orden de enseñanza establecido por el programador no se podía cambiar, es decir, si el alumno no había entendido claramente los contenidos expuestos hasta ese momento no tenía forma de volver atrás. Un tiempo después aparecen los programas ramificados con capacidad para actuar según la respuesta de los alumnos, [ALM05] aunque no ofrecían una enseñanza individualizada. A comienzo de los años 70 surgen los sistemas generativos bajo el principio de que el material de enseñanza podría ser generado por la misma computadora, ellos son capaces de generar problemas, construir sus soluciones, y diagnosticar las respuestas del alumno, controlando, a su vez, el nivel de dificultad de los problemas, pero estos sistemas no servían para todo tipo de enseñanza ya que la complejidad para generar problemas aumenta en determinadas áreas de trabajo.

Paralelamente en el área de la Inteligencia Artificial se comenzaron a construir sistemas que intentaban simular el razonamiento o lógica humana y a finales de los años 70 surgió una nueva rama en donde los investigadores intentaban desarrollar sistemas tutoriales inteligentes de instrucción asistida por ordenador. [SOL05] Estos se caracterizan por centrar el proceso de aprendizaje desde una perspectiva individualizada, es decir, los estudiantes aprenden mediante un proceso de interacción sistema-estudiante.

Un hito importante se produce con la aparición de las redes de computadoras pues estas ya no solo sirven como vehículo para hacer llegar a los estudiantes materiales para la autopreparación, sino para crear un entorno fluido y multimediático de comunicaciones entre los participantes del proceso de enseñanza-aprendizaje. [ROJ03], abren nuevos

horizontes en el campo de la educación, introduciendo flexibilidad en los procesos docentes.

Desde la década de los 90, Internet se convirtió en una herramienta fundamental de comunicación, información e integración, que permite a los usuarios ahorrar tiempo y dinero, además de tener a su alcance todos los productos y servicios que requieran sin fronteras de espacio y tiempo. [PER03]

En esta última etapa, el potencial que Internet ofrece como herramienta docente se extiende a diversas perspectivas:

- Como complemento de la docencia presencial, mediante el desarrollo de Web que contengan materiales de trabajo útiles para los alumnos.
- Como soporte de la gestión, mediante herramientas que faciliten las tareas tanto docentes (tutorías, sesiones de discusión, exámenes) como administrativas (listas de admitidos, programas docentes, calificaciones).
- Como aula virtual afectiva, llegando a crear en la red una infraestructura capaz de integrar todos los elementos docentes habitualmente empleados en la formación universitaria. [PER05]

Con el acceso a Internet se amplían las posibilidades de comunicación para los profesores y sus estudiantes a distancia quiénes pueden usar por ejemplo:

El correo electrónico que constituye un instrumento regular de trabajo de profesores, investigadores y estudiantes pues permite el intercambio sistemático, establecer listas de discusión y controlar el proceso de enseñanza aprendizaje.

Los foros: que hacen posible acceder a grupos de discusión organizados por temas, en todas las latitudes geográficas. El profesor puede establecer el foro del curso para promover la interacción entre estudiantes. La clase o las opiniones o reflexiones de los estudiantes pueden ser enviadas a otros participantes y estos pueden enviar sus comentarios o preguntas a toda la clase, y cada uno de los otros estudiantes puede responder libremente.

El Foro es una de las actividades colaborativas centrales en casi todas las plataformas interactivas. Esta actividad permite la reflexión colectiva sobre determinados temas y transmitir noticias a todos los participantes inscriptos en un curso. El Foro fue, en su tiempo, una fundamentación de las posibilidades de carácter pedagógico inherentes de Internet basándose en la teoría de la conversación.

Las video conferencias interactivas favorecen la comunicación por medio del video y audio bidireccional entre participantes en diversas localizaciones geográficas y tienen las ventajas de poder establecer contacto visual entre los estudiantes y el profesor o entre estudiantes localizados en sitios remotos; soporta el uso de diversos medios de enseñanza de comunicación: pizarras, documentos electrónicos, y videos que pueden incorporarse a la transmisión. [ROJ03]

El popular servicio Word Wide Web (WWW): Sistema de interconexión que combina imágenes, textos, sonidos, animaciones, programas y formularios para entregar información.

Las publicaciones electrónicas: Su contenido es recibido por todos los suscritos a la publicación, lo que les permite estar enterados de las principales novedades y actualizaciones en el sector o campo de esa publicación. Pueden ser boletines, revistas, etc.

Internet Relay Chat: Es uno de los servicios de conversación en línea que permite interconectar a personas de diferentes lugares del país y del mundo en torno a canales de conversación con temas de diversa índole. [PER03]

En opinión de Brown, citado en [VIZ98] hay dos aspectos que vale la pena resaltar de la Internet. En primer lugar, a diferencia de la radio y la televisión, permite la comunicación bidireccional, quien la usa no solo recibe información también es capaz de responder a ella por el mismo medio. En segundo lugar aún “los pequeños esfuerzos de muchos con los grandes esfuerzos de pocos”, permite contactar y obtener ayuda directa de los expertos en áreas específicas mientras a su vez permite que todos ayuden a otros con problemas más pequeños.

Existe hoy un movimiento que propone nuevas estrategias didácticas en las que se defiende que el aprendizaje de calidad, especialmente el aprendizaje de la ciencia, requiere cooperación, en este sentido el aprendizaje colaborativo permite al estudiante aprender a dar explicaciones sobre los sucesos que experimenta, justificar sus acciones y atender a las explicaciones de sus compañeros. Esto por un lado y las posibilidades de interacción con independencia de las distancias geográficas o de la concurrencia del tiempo que brindan hoy las tecnologías de la información y las comunicaciones han impulsado la evolución de los sistemas de enseñanza asistida por computadora hacia nuevos escenarios de enseñanza - aprendizaje, los cuales se conocen como entornos virtuales de enseñanza, ambientes de enseñanza – aprendizaje asistidos por computadoras, hiperentornos educativos, etc., que permiten implementar e impartir cursos en la Web, cursos en línea o cursos virtuales.

Por último y no por ello menos importante, las técnicas de realidad virtual han permitido “transportar” al estudiante a un mundo ideal que refleja la realidad que se estudia lo cual tiene un innegable valor práctico para estudiar fenómenos riesgosos o difíciles de observar. [LEZ98]

1.1.1 Enfoque de la Informática como medio de enseñanza

Hoy en día, la Informática como medio de enseñanza cuenta con una amplia gama de tipos de programas que pueden ser empleados con múltiples enfoques. Cada uno de estos enfoques tiene propósitos específicos, dirigidos a contribuir con el desarrollo de diferentes funciones del proceso docente, por lo que es usual encontrar en la literatura clasificaciones como la siguiente: Tutoriales, Entrenadores, Repasadores, Evaluadores, Simuladores, Libros Electrónicos, Juegos Instructivos, etc. [PSN01]

Por su parte Galvis [GAL94] en su obra “Ingeniería de software educativo” los clasifica en: sistema tutorial, sistema de ejercitación y práctica o entrenadores, simulador, juego educativo, micromundo exploratorio, sistemas expertos y sistemas inteligentes de enseñanza/aprendizaje.

Sevillano en su obra “Nuevas Tecnologías, Medios de Comunicación y Educación” [SEV98] expone una clasificación de acuerdo al uso del software educativo, refiriéndose a: simulaciones, realidad virtual, los juegos de roles, laboratorios, telecomunicaciones, procesadores de textos, bases de datos, hojas de cálculo, paquetes estadísticos y diseño de gráficos.

Según [MED07], el grupo de Informática Educativa de la Universidad Central “Marta Abreu” de Las Villas (UCLV) comparte el criterio de clasificación de software educativo de acuerdo a:

- a) **Función educativa:** sistema tutorial, sistema de ejercitación y práctica (entrenador), simulaciones y juegos didácticos.
- b) **Forma de presentación:** multimedia, hipermedia [GAR00] y sitios Web.
- c) **Uso o no de técnicas de IA:** sistemas convencionales, sistemas inteligentes y estos sistemas a su vez pueden ser del tipo a) o b).

Estas clasificaciones no son excluyentes, en un mismo software educativo las mismas pueden combinarse.

1.2 Los Agentes Inteligentes

La Inteligencia Artificial (IA) es una rama de la Ciencia de la Computación que se dedica al estudio de cómo lograr que las computadoras y los softwares puedan simular el comportamiento humano y realizar funciones que actualmente los humanos realizan mejor. [RUS95] Entre estas funciones se encuentran razonar, aprender y procesar el lenguaje natural. También tiene dentro de sus objetivos darle solución a problemas complejos donde no existe un algoritmo para su solución o de existir, la complejidad computacional es muy alta y es físicamente imposible poder ejecutarlo. [RUS95]

Para ello la IA cuenta con diversas vertientes como la robótica, los sistemas expertos, las redes neuronales, los juegos y los Agentes Inteligentes (AI).

De manera general, agrupando las definiciones de autores como [RUS95][BAR99][HER99][MAE94], puede considerarse un AI como una entidad que se encuentra en un ambiente e interactúa con él, que es capaz de tener autonomía y control de sí mismo, que puede actuar solo ante cambios en este ambiente sin perder el objetivo para el cual fue creado, que es capaz de interactuar con otros agente, brindarle ayuda, de hacer lo correcto y de aprender de su experiencia, teniendo en cuenta que a medida que este aprendizaje sea mayor, mayor será su grado de autonomía porque podrá actuar basado en su experiencia ante un suceso en el presente. Puede también si su función lo requiere viajar por toda la red y explorarla retornando a su origen, ya que no terminan ni se agotan, pueden trabajar eternamente.

Varios de los autores [GRE97][HAY02][HER99] toman como referencia las aplicaciones en las que pueden ser empleados los AI, coincidiendo la mayoría de ellos en sus clasificaciones, así pueden considerarse los siguientes tipos de agentes: estáticos, móviles, deliberativos, reactivos, de interfaz de usuario, de información y colaborativos o distribuidos.

Para diseñar un buen agente hay que pensar en su arquitectura. La arquitectura de un agente define los mecanismos que permiten interconectar los componentes tanto de software como de hardware, que hacen que el agente se comporte como tal.

Cada arquitectura particular para construir agentes, especifica como se descompone en un conjunto de módulos que interactúan entre sí para lograr la funcionalidad requerida. Un aspecto básico que diferencia una arquitectura de otra es el método de descomposición del agente en tareas particulares.

La arquitectura es considerada por algunos autores como la manera en que se implementa el proceso de toma de decisiones de los agentes. Se pueden dividir en cuatro tipos [MAE94]:

- Agentes basados en lógica

Estos agentes llevan a cabo el proceso de toma de decisiones mediante deducción lógica. El enfoque tradicional simbólico sugiere que la conducta inteligente puede ser generada en un sistema con representación simbólica de su ambiente y de su

conducta deseada con una manipulación sintáctica de dichas representaciones. La representación simbólica está constituida por fórmulas lógicas y la manipulación sintáctica que se corresponde con la deducción lógica o la prueba de teoremas.

Los Agentes Deliberativos usan una arquitectura basada en lógica. De manera general los agentes implementados con esta arquitectura tienen elegancia y semántica limpia, aunque durante el proceso de toma de decisiones el ambiente no debe cambiar y poseen alta complejidad computacional por lo que no operan bien en ambientes restringidos por el tiempo.

- Agentes reactivos

El proceso de toma de decisiones es mediante un mapeo directo donde dada una situación se obtiene la acción. Es un nuevo enfoque contrario al simbólico. Aquí los agentes construidos si pueden operar en ambientes restringidos por el tiempo.

- Agentes con intereses, deseos y creencias.

El proceso de toma de decisiones consiste en manipular estructuras de datos que son las que representan los intereses, deseos y creencias del agente.

- Capas

Se usan capas de software ordenadas de mayor a menor según la capacidad de razonamiento o nivel de abstracción.

Las arquitecturas descritas anteriormente, tienen gran aplicabilidad actualmente, ya que el proceso de toma de decisiones es muy importante en un AI, pues influye directamente en su autonomía.

El LPA WIN-PROLOG o el Ciao-PROLOG permiten crear las bases de conocimiento que los agentes utilizan para el aprendizaje, ya que tienen las ventajas de este lenguaje para la representación del conocimiento y su procesamiento. [SHA05]. El WIN-PROLOG provee una biblioteca de agentes que cuenta con dll y funciones para este fin.

Por otro lado, las tecnologías más novedosas están en el campo del procesamiento y presentación de la información. Y no existe la información si el agente no es capaz de

entenderla, por eso se están definiendo estándares para intercambio de información que permitan la interoperación entre agentes.

El ACL -Agent Communication Language- es un lenguaje de comunicación de agentes desarrollado por ARPA, que tiene tres componentes: un vocabulario, un lenguaje de contenido llamado formato de intercambio de información KIF -Knowledge Interchange Format- y un lenguaje de comunicación llamado KQML -Knowledge Query Manipulation Language- (Finin, 1994). Un mensaje de ACL es un mensaje en KQML que consiste de una directiva de comunicación y de un contenido semántico en KIF expresado en términos del vocabulario [BAR99].

Para que esa información afecte al estado interno y a las acciones del agente en el futuro, el agente debe de ser capaz de adaptarse. Las redes neuronales son una tecnología que permite asociar entradas a salidas, y adaptarlas para dar las salidas deseadas. También la lógica fuzzy permite resumir el conocimiento en una serie de reglas que expresan la información con los mismos valores lingüísticos que lo hacen los seres humanos.

Por otro lado, los grupos de agentes pueden también evolucionar usando algoritmos genéticos, que podrían actuar sobre agentes de la misma forma que la evolución: eliminando los peores y dejando que los mejores se reproduzcan, cambiando a su vez con la posibilidad de crear algo mejor.

1.2.1 Los Agentes Inteligentes en ambientes de enseñanza aprendizaje

Los AI han sido llevados al campo de la enseñanza de forma exitosa, dotando a los sistemas instruccionales de características distintivas de gran utilidad como el manejo de emociones, el apoyo efectivo del trabajo colaborativo y las interacciones pedagógicas adaptativas. De este modo es posible hablar de interfaces que pueden tomar la información y representarla con discurso, con gesto, con expresión facial, con postura, etc.

Los sistemas basados en Agentes Inteligentes pueden resultar ventajosos como medios de instrucción, entre otras razones, porque: [JUL00]

- Los estudiantes pueden realizar preguntas en cualquier momento.
- El agente puede estar siempre “observando” el actuar del estudiante.
- El agente puede reconstruir y redefinir su actuar en cada momento a partir del actuar del estudiante.
- El agente puede adaptar su actuar a situaciones inesperadas.
- El estudiante puede tomar el control en cualquier momento.
- En caso de errores el agente ayuda a que el estudiante aprenda de ellos.
- En cualquier momento, el estudiante puede consultar en que parte del proceso se encuentra y las tareas faltantes.

La tecnología de AI, lleva desarrollándose un poco más diez años, y ha sido motivo de interés de grandes grupos de investigación y de personalidades del área de la IA. Los agentes pedagógicos no han sido muy desarrollados pero significativos avances sí han tenido lugar.

Un agente pedagógico es un agente que toma decisiones acerca de cómo maximizar el aprendizaje de un alumno, y el “entorno” que este observa en su proceso de aprendizaje; para cumplir con sus metas, un agente pedagógico puede actuar como un tutor virtual, como un estudiante virtual, o como un compañero de aprendizaje que ayuda al estudiante. Un área de aplicación especialmente relevante de estos agentes es el entrenamiento de equipos de trabajo. Esta aplicación plantea retos interesantes como son el hecho de tener que tutelar simultáneamente a más de un estudiante, o el hecho de tener que manejar simultáneamente conocimiento “espacial” acerca del entorno y conocimiento “procedimental” acerca de las operaciones que pueden ser realizadas en el mismo [GON03]

COACH es un tutor inteligente creado en 1994. Proporciona asistencia personalizada a estudiantes utilizando la computadora para aprender sobre dominios específicos tales como el lenguaje de programación LISP o el Sistema Operativo Unix

ADELE es un Agente para Ambientes de Aprendizaje a Distancia (Agent for Distance learning Environments) desarrollado en 1997. Es un agente pedagógico que corre en cada una de las computadoras e interacciona con cada uno de los estudiantes cuando ellos navegan en los materiales del curso a través de la web. ADELE es responsable de monitorear al estudiante y recoger las acciones de él para adaptar la presentación del curso. Reporta el comportamiento de cada uno de los estudiantes al servidor central al terminar la sesión.

PPP Persona es un Plan basado en Presentaciones Personalizadas creado en 1998. Es un agente pedagógico animado para presentaciones interactivas WWW. La persona aparece en diferentes formas: figuras animadas o modelos 3D. EL PPP Persona sigue el aprendizaje a través de materiales Web usando presentaciones donde dibuja los elementos de las páginas Web y proporciona comentarios sintetizados.

CU ANIMATE desarrollado en el 2002. Consiste en un conjunto de herramientas de software para habilitar conversaciones con caracteres animados. Los agentes animados están incorporados en un ambiente de aprendizaje interactivo libro-multimedia para ayudar a los niños a adquirir conocimiento a través de la lectura.

Otros avances en esta línea, son sistemas que animan a los niños a desarrollar y contar sus historias, desarrollando la capacidad de representar pensamientos simbólicamente y de compartirlos con otros niños en su propio lenguaje. Uno de estos desarrollos, el “Storytelling” es un sistema que apoya el trabajo en forma colaborativa entre niños, proporcionando un ambiente para que estos aprendan las habilidades más importantes del lenguaje. El agente conversacional SAM es otro resultado importante, este puede contar historias en forma colaborativa con los niños, está pensado para ser un niño más del grupo.

Por su parte, [CHO02] se refiere al desarrollo de agentes para motivar a diferentes grupos, por ejemplo agentes que enseñan en procesos de aprendizajes sociales, agentes que son alborotadores e incluso proporcionan opiniones erróneas con la intención de entregar un elemento disonante entre lo que el estudiante sabe y lo que el agente sugiere, de manera que obligan al estudiante a buscar nueva información para reducir la disonancia.

Las aplicaciones desarrolladas aprovechan las fortalezas de los agentes y brindan solución a problemas muy complejos donde las técnicas existentes son muy difíciles de implementar y/o mantener. Por otro lado, muestran cuanto se ha evolucionado desde la perspectiva de la concepción de crear sistemas donde se contemplaba solamente la interacción del alumno con el ordenador a considerar en los nuevos sistemas, el entorno social que afecta al aprendizaje del alumno mediante el ordenador. [GON03]

1.3 Sistemas Adaptativos

Las corrientes actuales de SEBW se enfocan a implementar diversos paradigmas de trabajo como son la enseñanza centrada en el estudiante, el aprendizaje colaborativo y la educación adaptativa.

Según Benyon [BEN94], un sistema adaptativo “es aquel que, basado en el conocimiento, altera automáticamente aspectos de funcionalidad e interacción para lograr acomodar las distintas preferencias y requerimientos de sus distintos usuarios”. Como ejemplos de comportamiento adaptativo podemos citar la presentación de formularios y menús dependiendo de la tarea a realizar, la presentación de la información relevante según la tarea o usuario que la demande, o el ofrecimiento de ayuda según el contexto de trabajo.

Son muchos los beneficios que se esperan de los sistemas adaptativos, algunos de ellos son:

- Aumentar su ciclo de vida, facilitando su mantenimiento.
- Extender el rango de usuarios, desde el novato al experto.
- Satisfacer las demandas del usuario, reduciendo temores y aumentando el atractivo y la flexibilidad, logrando así una mejor aceptación.

En la literatura revisada se encontraron sistemas adaptativos para diferentes dominios, predominan los de propósito general, los destinados a sistemas de información on-line y los creados para la enseñanza.

La mayoría de estos sistemas asume un modelo de usuario y define la adaptación basada en este modelo de usuario. En el caso de los sistemas para dominios educacionales el modelo del usuario es el modelo del estudiante.

La educación adaptativa incorpora la Inteligencia Artificial para que el propio sistema adapte los servicios y material que ofrece al aprendiz, por medio del modelo del estudiante que desarrolla y actualiza a lo largo del proceso en donde se representan los intereses particulares del individuo, sus aptitudes, desempeño y resultados; de tal forma que a lo largo del proceso de enseñanza se le ofrezca una atención personalizada que estimule su aprendizaje [OKA00] [PEÑ04].

Los Sistemas Hipermedia Adaptativos son una opción interesante para ello pues su objetivo es que sea el sistema el que se adapte al usuario y no al contrario, como sucede en los hipermedia “clásicos”, los cuales muestran el mismo contenido y los mismos enlaces a todos los usuarios. Por lo tanto se constituyen como una alternativa al enfoque “una-talla-para-todos” en el desarrollo de sistemas hipermediales [BRU96] [BER04]. Su naturaleza permite configurar entornos educativos para conseguir que los alumnos alcancen los objetivos de aprendizaje establecidos mediante contenidos y recorridos adecuados a sus aptitudes, intereses y preferencias [BER04].

Para lograr este tipo de sistemas, según los autores antes citados, se utilizan básicamente tres modelos. El del dominio que establece el conocimiento que se desea transmitir, el del usuario (modelo del estudiante) que representa los objetivos y particularidades de cada alumno y por último el de adaptación. (ver Fig. 1.1)

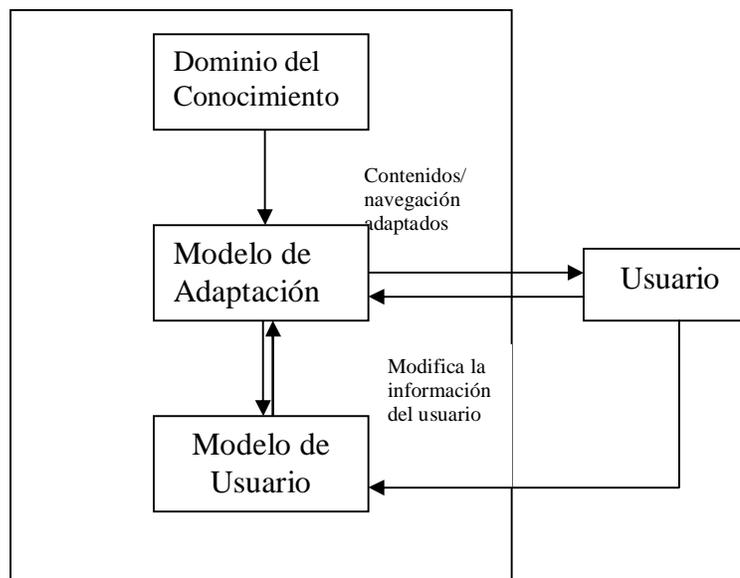


Fig. 1.1: Componentes de un SHA.

Modelo del usuario.

Para ofrecer un servicio adecuado a cada usuario es necesario que el sistema cuente con una representación de sus características propias y en base a dicha representación, tome las decisiones pertinentes en la interacción [FIS00] [COS05] [MED07a].

El modelo de usuario tiene como objetivo representar la relación de cada sujeto con el conocimiento que se le desea transmitir. Para ello, almacena aspectos relevantes de cada usuario, como sus preferencias, conocimientos previos, intereses y otros que utiliza para realizar la adaptación.

En el caso de los Sistemas de enseñanza aprendizaje adaptativos (SEAA) se habla del modelo del estudiante (ME) para contener información acerca del aprendiz, incluyendo el nivel del conocimiento que ha adquirido, el grado de habilidades que ha desarrollado, la capacidad para la realización de las tareas, la facilidad de aprendizaje y su orientación personal. Esta información se puede adquirir de varias formas: mediante pruebas, diagnósticos y registro de las actividades durante las sesiones, por lo que el modelado es un proceso dinámico [PEÑ04a].

Existen diferentes modelos de estudiantes tales como: modelo del estudiante abierto, modelo del estudiante activo, modelo del estudiante cooperativo y modelo del estudiante en grupo.

Modelo del dominio.

El objetivo de este modelo es estructurar el dominio del conocimiento que se desea transmitir, posibilitando registrar, organizar y proveer acceso a los materiales de aprendizaje desarrollados para el sistema, facilitando la creación de contenidos por parte de diferentes autores en forma individual o colaborativa y permitiendo la visualización del contenido provisto al estudiante.

La estructura del dominio de conocimiento se puede representar de diferentes maneras, por ejemplo mediante grafos de conceptos, grafos de requisitos, árboles jerárquicos, y mapas conceptuales, entre otras (ver Fig. 1.2).

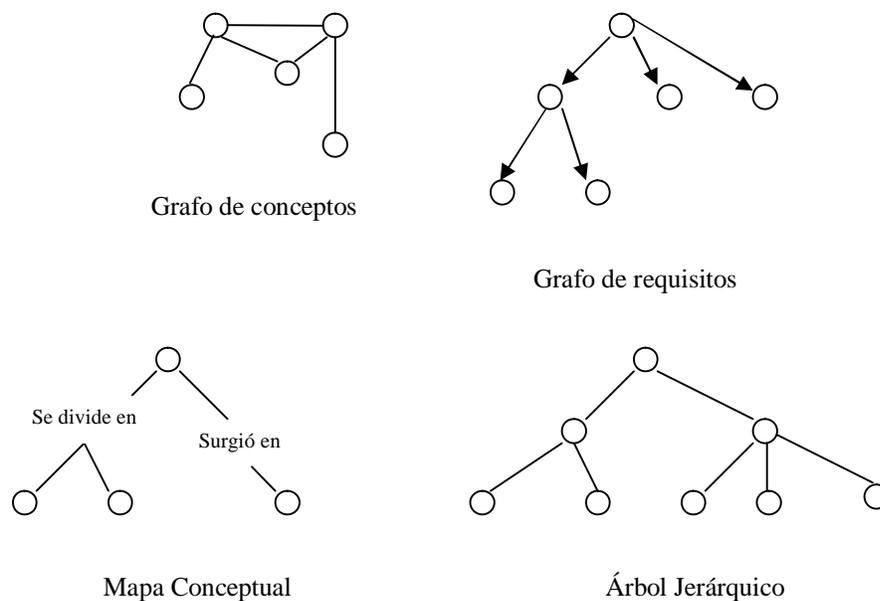


Fig. 1.2: Algunas formas para definir la estructura del dominio del conocimiento.

Modelo de Adaptación.

El modelo de adaptación contiene la descripción de la funcionalidad adaptativa del SHA. Lo que incluye la adaptación del contenido, los enlaces y las actualizaciones al modelo de usuario. Generalmente, la adaptación se lleva a cabo mediante reglas que especifican qué y cómo se deben mostrar y comportar los elementos del sistema teniendo en cuenta el modelo del usuario.

[HEN03] divide las reglas en aquellas que describen la funcionalidad adaptativa (sugieren un documento, generan recorridos, etc.) y en las que definen el tratamiento adaptativo que producirá en tiempo de ejecución (por ejemplo, ordenar los enlaces según su utilidad para un usuario en particular, establecer anotaciones en los documentos, etc.).

1.3.1 Elementos a tener en cuenta para lograr la adaptación

Deagostini y Cormenzana en su artículo: Interfaces de Usuario Inteligentes: Sistemas Adaptativos, plantean que para comprender mejor como estos sistemas se adaptan al usuario se pueden considerar dos aspectos fundamentales: *la dinámica*, que corresponde a los objetos que pueden ser adaptables y *la métrica*, usada para diseñar, construir y evaluar los sistemas adaptativos. Existen varios ejemplos de componentes dinámicas, algunos de ellos son: las ayudas, el lenguaje de comandos y los mensajes de error siempre y cuando sean explícitos y adecuados.

Como ejemplo de métricas señalan aquellas que responden a las siguientes preguntas:

- ¿Qué queremos mejorar al adaptarnos según el fin considerado?
- ¿Qué aspectos suponemos que tenemos que modificar indirectamente para lograr los objetivos que directamente no podemos alcanzar?
- ¿Qué señales, y en qué forma desencadenan los cambios o acciones adaptativas?

- ¿Qué elementos del diálogo o la interfaz cambian debido al desencadenamiento?
- ¿Cuánto cuesta el proceso en términos de performance del sistema?

Métodos y Técnicas de adaptación.

[BRU96] considera que existen métodos y técnicas de adaptación. Un método es una generalización de una o varias técnicas de adaptación, es decir un mismo método puede implementarse con varias técnicas. Estas, en cambio, son procedimientos para lograr que un SHA se adapte. A continuación se detallan los métodos y técnicas correspondientes a la presentación adaptativa y las que sirven de soporte a la navegación adaptativa.

Presentación Adaptativa.

La presentación adaptativa provee al usuario de prerequisites, explicaciones comparativas o adicionales, da alternativas de explicación (presentar información de diferentes maneras), y reordena la información según el modelo de usuario [DEB99].

Técnicas: Las técnicas de la presentación adaptativa se encargan de manipular el contenido para adecuarlo a las características del usuario. La mayoría de las técnicas se usan para adecuar el texto, pero también se pueden emplear en contenidos multimedia [BER04].

Soporte a la navegación adaptativa.

Métodos: Los métodos de la Navegación adaptativa se encargan de ayudar a los sujetos cuando estos navegan por el hiperespacio, guiándolos, orientándolos o facilitándoles la posibilidad de crear vistas personalizadas a sus intereses.

Técnicas: Las técnicas del soporte a la navegación adaptativa manipulan los enlaces que contienen las páginas web para representar información relevante y apropiada para cada sujeto.

1.3.2 Sistemas Adaptativos en la Enseñanza

Como ya se dijo antes en este informe, los Sistemas Adaptativos resultan ventajosos para crear ambientes de enseñanza aprendizaje para ser usados tanto en actividades presenciales como a distancia. En la literatura revisada se encontraron disímiles ejemplos a continuación se comentarán algunos de ellos.

ELM-ART (Episodic Learner Model-Adaptive Remote Tutor) es un sistema tutor inteligente basado en Web que apoya el aprendizaje de la Programación en Lisp. Es el resultado de la evolución de ELM-PE (Episodio Learning Model – Program Environment) sistema con el cual los alumnos del curso de introducción al lenguaje de Programación Lisp en la Universidad de Trier (Alemania) completaban sus clases presenciales realizando ejercicios, consultando ejemplos, etc.

ELM-ART organiza su dominio de conocimiento a partir de un glosario y un libro de texto electrónico indexado. Cada entrada del glosario corresponde a un concepto del dominio proveyendo una descripción y proporcionando enlaces a cada sección relacionada. [BER04] El contenido del libro de texto se estructura jerárquicamente en capítulos, secciones, subsecciones y elementos. Cada nivel Terminal puede contener una presentación atómica del contenido, ejemplos, problemas o evaluaciones.

El modelo del estudiante lo representa a través de un modelo Overlay que computa los estados educativos de cada concepto y genera la anotación de enlaces.

InterBook.

InterBook [BRU98] [SCH96] [EKL97] empezó a desarrollarse en 1995 siguiendo un enfoque basado en conceptos. Se trata de un sistema de autor que implementa la arquitectura definida en ELM-ART para construir materiales educativos en línea mediante la metáfora del libro, presentándolos como un conjunto de libros de texto electrónicos adaptativos sobre el mismo tema.

Para representar el modelo del dominio, en su forma más simple, emplea solo conceptos (como temas, elementos de aprendizaje, objetos, etc) definidos como piezas elementales del conocimiento sobre el dominio tratado; en su forma más avanzada

utiliza una red de nodos interconectados mediante enlaces. Los nodos representan los conceptos del dominio, mientras que los enlaces reflejan las diferentes clases de relación que existe entre dichos conceptos.

Cada libro se estructura a partir de su contenido y de un glosario, cada entrada al glosario se corresponde a una descripción de cada concepto, por lo tanto, al igual que ELM-ART, se trata de un glosario y un índice integrado.

En InterBook cada libro de texto electrónico se estructura jerárquicamente en unidades de información en forma de capítulos, secciones, subsecciones y elementos. Cada nivel Terminal es una presentación atómica (ejemplo, problema o evaluación). Con el fin de proveer inteligencia a las unidades del libro, estas se indexan con los conceptos. Las unidades terminales se asocian a una lista de conceptos relacionados que especifica para cada concepto su nombre y su tipo de rol, que puede ser resultado o prerequisite.

Finalmente, diversos libros electrónicos sobre un mismo tema se agrupan formando una estantería de libros e indexándose con los mismos conceptos. Ello con el objetivo de que los alumnos utilicen varios libros a la vez y realicen búsquedas.

1.4 Fundamentación de la propuesta de diseño del Entorno de aprendizaje

El hombre llega a conocer su entorno material y social, del cual es parte o elemento inseparable, mediante un complejo proceso de aprendizaje el cual lo educa y capacita para interactuar con ellos de manera lógica y dialéctica [DEF09]. La Pedagogía ha evolucionado mucho desde su origen etimológico que significaba conducir o llevar a un niño en el sentido espiritual o enseñarlo. Es una ciencia de carácter psicosocial que tiene por objeto el estudio de la educación con el fin de conocerlo y perfeccionarlo. También es una ciencia de carácter normativo porque no se dedica a describir el fenómeno educacional sino a establecer las pautas o normas que hemos de seguir para llevar a buen término dicho fenómeno [WIK09].

Estas pautas o normas no son más que las tendencias pedagógicas. Con ellas se normalizan las formas del trabajo que se ha de desarrollar en el proceso de enseñanza facilitador de un aprendizaje eficiente, el papel que se le asigna tanto al educando como al educador. Es por eso que debemos tener en cuenta la tendencia pedagógica que se asumirá a la hora de diseñar un software educativo. A continuación se hace un pequeño estudio de la misma.

1.4.1 Enfoque histórico-cultural

La Pedagogía cubana sigue la concepción del enfoque histórico-cultural de Vigotsky [VIG78] enriquecido por sus seguidores extranjeros y cubanos, lo que implica que a partir del carácter rector de la enseñanza para el desarrollo psíquico, considera a esta como fuente de ese desarrollo. Lo central en el proceso de enseñanza consiste en estudiar la posibilidad y asegurar las condiciones (sistemas de relaciones, tipos de actividad) para que el estudiante se eleve mediante la colaboración, la actividad conjunta, a un nivel superior. Partiendo de lo que aun no puede hacer solo, llegar a lograr un dominio independiente de sus funciones.

Para Vigotski el aprendizaje es una actividad social, y no sólo un proceso de realización individual como hasta el momento se había sostenido; una actividad de producción y reproducción del conocimiento mediante la cual el alumno asimila los modos sociales de actividad y de interacción, y más tarde en la escuela, además, los fundamentos del conocimiento científico, bajo condiciones de orientación e interacción social [TEO09].

Este concepto del aprendizaje pone en el centro de atención al sujeto activo, consciente, orientado hacia un objetivo; su interacción con otros sujetos (el profesor y otros estudiantes), sus acciones con el objeto, con la utilización de diversos medios en condiciones socio históricas determinadas. Su resultado principal lo construyen las transformaciones dentro del sujeto, es decir, las modificaciones psíquicas y físicas del propio estudiante, mientras que las transformaciones en el objeto de la actividad sirven sobre todo como medio para alcanzar el objetivo de aprendizaje y para controlar y evaluar el proceso [DEF09].

En lo relativo al estudiante, implica utilizar todos los resortes de que dispone la personalidad (su historia académica, sus intereses cognoscitivos, sus motivos para el estudio, su estado emocional) en relación con los que aporta el grupo de clase involucrando a los propios estudiantes en la construcción de las condiciones más favorables para el aprendizaje.

Desde el punto de vista del profesor, supone extraer de sí mismo, de su preparación científica y pedagógica todos los elementos que permitan el despliegue del proceso de redescubrimiento y reconstrucción del conocimiento por parte del estudiante, de sus particularidades personales, la relación de comunicación en sus distintos tipos de función (informativa, afectiva y reguladora) que permita un ambiente de cooperación y de colaboración, de actividad conjunta dentro del aula [DEF09].

Según Vigotsky [VIG78] lo que las personas pueden hacer con la ayuda de los otros puede ser, más indicativo de su desarrollo mental que lo que pueden hacer por sí solos, de allí la importancia que se adjudica a la actividad conjunta, a la relación de cooperación entre los alumnos, y entre éstos y el profesor. Esta concepción cambia la tradicional relación entre autoridad y distancia existente entre ambos participantes del proceso, señala como función fundamental del profesor la orientación y guía del estudiante, con el fin de potenciar sus posibilidades y convertir en realidad las potencialidades de su Zona de Desarrollo Próximo (ZDP). Este es uno de los conceptos esenciales en la obra de Vigotsky y no es otra cosa que la distancia entre el nivel real del desarrollo y el nivel de desarrollo potencial, determinado a través de la resolución de un problema bajo la guía de un adulto o en colaboración con un compañero más capaz.

Partiendo del enfoque histórico cultural y la teoría de la actividad es posible explicar claramente cómo el proceso de aprendizaje se debe convertir en el centro de atención, a partir del cual se proyecte el proceso pedagógico, [DEF09] lo cual se pone de manifiesto en el sistema que se propone en este trabajo, el cual tiene fines pedagógicos, pues pretende facilitar el proceso de enseñanza-aprendizaje en la asignatura Programación Lógica. La inserción de agentes inteligentes en APA-Prolog permitirá personalizar la atención a los estudiantes. Este ambiente basado en mapas conceptuales le permitirá al estudiante utilizar lo disponible en el sistema de relaciones más cercano

a él para propiciar su interés y un mayor grado de participación e implicación personal en las tareas de aprendizaje.

Debido al perfil de los estudiantes de Ingeniería Informática, la utilización de las computadoras y, en general de las Tecnologías de la Informática y las Comunicaciones (TIC), para su estudio y apoyo al aprendizaje es algo básico y fundamental, disponen de computadoras en su sistema de relaciones más cercano, en este caso en el laboratorio de Informática perteneciente al CUSS. Además debemos tener en cuenta que la computadora es un poderoso instrumento, pues brinda la oportunidad de transformar la pedagogía haciéndola más efectiva y emotiva, a la vez que mejora y fortalece el papel educador del profesor.

La asignatura Programación Lógica rompe con el paradigma de programación que traen los estudiantes hasta el momento, por tanto son temas difíciles de comprender. Si se proporciona adaptabilidad a APA-Prolog utilizando agentes inteligentes se contribuye a personalizar la atención al estudiante mostrando la información y los recursos adecuados.

1.5 Selección de la tecnología adecuada

El software libre es una alternativa para la excesiva globalización corporativa que tiene la potencialidad de acabar con la monopolización de la tecnología para siempre y puede llevar a un nuevo modelo de economía en este sentido [ART09]. La trascendencia del uso de las herramientas de software, prácticamente en todos los campos de la actividad humana, crea la necesidad de que éstas sean consideradas como un bien cultural esencial para el conjunto de la humanidad. El software libre, allana las diferencias y pone a todos en igualdad de condiciones.

La definición de "Software Libre" surge en 1984 a partir del esfuerzo de un grupo de entusiastas liderados por Richard Stallman, con el objetivo de eliminar las restricciones sobre el copiado, redistribución, entendimiento y modificación de programas de

computadoras, así como recuperar el ambiente de camaradería y cooperación que existía en los primeros días de la computación.

“Software Libre” se refiere a la libertad de los usuarios para ejecutar, copiar, distribuir, estudiar, cambiar y mejorar el software. De modo más preciso, se refiere a cuatro libertades de los usuarios del software [POR09]:

- ✓ La libertad de usar el programa, con cualquier propósito.
- ✓ La libertad de estudiar cómo funciona el programa y adaptarlo a sus necesidades.
- ✓ La libertad de distribuir copias.
- ✓ La libertad de mejorar el programa y hacer públicas las mejoras a los demás, de modo que toda la comunidad se beneficie. El acceso al código fuente es un requisito.

El propósito de realizar software libre no es ser popular, ni nunca lo ha sido; sino crear software bueno técnicamente y además que no afecte la libertad de las personas. Una vez que un producto de software libre ha empezado a circular, rápidamente está disponible a un costo muy bajo o sin costo. Al mismo tiempo, su utilidad no decrece. Esto significa que el software libre se puede caracterizar como un bien público en lugar de un bien privado.

El uso de software libre evita la dependencia tecnológica, fundamentalmente de empresas de países desarrollados, permitiendo la generación de riquezas y recursos nacionales, que quedarían dentro del país. También evita el alto costo que supone el pago de las licencias de software, lo cual eleva el costo total de propiedad (TCO Total Cost of Ownership) de las computadoras.

Actualmente nuestro país se encuentra inmerso en una gigantesca migración hacia el software libre. El bloqueo que se nos impone limita, entorpece y encarece la adquisición del software propietario necesario para el desarrollo de nuestra sociedad, por esta razón la perspectiva del software libre se abre como solución a esta problemática para el desarrollo de la informática cubana.

1.5.1 Las Tecnologías Web

Las tecnologías Web permiten acceder a los recursos de conocimiento disponibles en Internet o en las intranets utilizando un navegador. Están muy extendidas por muchas razones: facilitan el desarrollo de sistemas de Gestión del Conocimiento; son flexibles en términos de escalabilidad, es decir, a la hora de expandir el sistema; resultan fáciles de usar e imitan la forma de relacionarse las personas, al poner a disposición de todos el conocimiento de los demás, por encima de jerarquías, barreras formales u otras cuestiones.

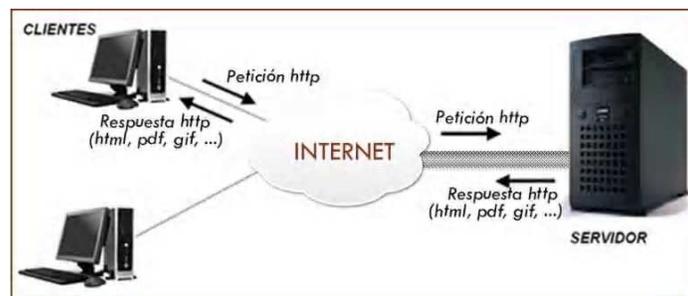


Fig. 1.3: Arquitectura de las tecnologías Web.

Una aplicación con una capa de presentación elaborada para ser consumida desde un navegador Web, permite ser usada en cualquier sistema operativo, convirtiéndose en una aplicación multiplataforma. Además, trae consigo la facilidad de uso, versatilidad y velocidad de interfaz [YUE03].

La capacidad para actualizar y mantener aplicaciones Web sin distribuir ni instalar software en las computadoras de los clientes es la razón clave de su popularidad. Además, alojando la aplicación en un servidor Web dedicado, se logra una disponibilidad total del producto.

Al ser la navegación Web tan popular no se necesita entrenamiento adicional por parte de los clientes para poder manipular la interfase de la aplicación.

1.5.2 ZOPE como servidor de aplicaciones Web

Una de las primeras definiciones de ZOPE dice así: “ZOPE es la nueva generación de servidores globales de aplicaciones. Enmarcado en forma de portal Web, programado mediante Python, con absoluta independencia de la plataforma en la que se ejecute y con una muy activa comunidad de usuarios detrás que soportan su carácter Open source, ZOPE es mucho más que un servidor Web” [ZOP08].

Lo que ZOPE nos proporciona es una plataforma de desarrollo Web robusta, claramente orientada a objetos y que soluciona gran parte de los problemas cotidianos de los desarrolladores de aplicaciones Web, facilitando enormemente su trabajo. Así, ZOPE garantiza automáticamente, por citar algunos ejemplos: integridad en las bases de datos, persistencia, control de acceso y herramientas de búsqueda integradas. ZOPE posee además una separación bastante clara entre lógica, datos y presentación. ZOPE se distribuye con una licencia Open Source y está escrito en Python [ZOP08].

ZOPE posee un conjunto propio de elementos que se vinculan con los clientes, las bases de datos, los sistemas de archivos y los servidores Web (ver Fig. 1.4).

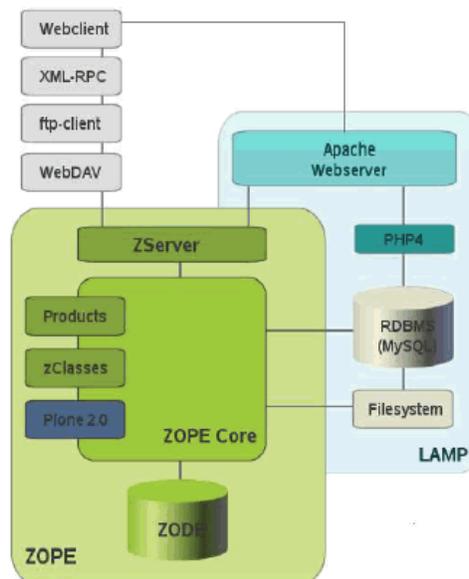


Fig. 1.4: Componentes de un SHA.

Principales características: código abierto, multiplataforma, incluye servidores propios (HTTP, WebDAV y FTP), se divide en tres niveles: datos, lógica de negocio y presentación, gestión de usuarios integrada en la aplicación, reutilización de código, alto nivel y soporte comercial.

ZOPE ayuda potencialmente a crear sitios Web con menos costos y mucho mas rápido que con otros servidor de aplicaciones. Esto es debido a sus características [ZOP09]:

- ✓ ZOPE es gratuito y es distribuido bajo licencia de software libre. A cambio de las muchas aplicaciones no libres comerciales las cuales son relativamente costosas.
- ✓ ZOPE permite a los desarrolladores crear aplicaciones Web con el solo uso de un navegador Web. Puede ser Mozilla, Internet Explorer, Netscape, OmniWeb, Konqueor, Opera, son todos compatibles para mostrar y manejar el entorno de desarrollo de ZOPE. ZOPE también permite a los desarrolladores delegar funciones de desarrollo de la aplicación a otros desarrolladores a través de Internet de una manera muy segura usando la misma interfaz. Muy pocos de los demás servidores de aplicaciones proponen este nivel de funcionalidad.
- ✓ ZOPE provee un granular y extensible entorno de desarrollo. Se puede integrar fácilmente ZOPE con diversos sistemas de autenticación y autorización como LDAP, WindowsNT, y RADIUS simultáneamente, usando módulos prefabricados. Muchas de los otros servidores de aplicaciones solo ofrecen algunos de estos sistemas de autenticación y autorización.
- ✓ ZOPE permite que equipos de desarrolladores se colaboren entre sí efectivamente. Entornos colaborativos requieren de herramientas que permitan a los usuarios trabajar sin interferir a otro, por eso ZOPE tiene: deshacer, versiones, historial y otras herramientas que ayudan a trabajar seguros y recuperarse de los errores. Muchos de los otros servidores de aplicaciones no ofrecen este tipo de características.
- ✓ ZOPE corre en las plataformas de sistemas operativos mas difundidas: Linux, Windows NT/2000/XP, Solaris, FreeBSD, NetBSD, OpenBSD y Mac OS X. ZOPE corre también en Windows 98/Me (recomendado solo para propósitos de

desarrollo). Muchos de los demás servidores de aplicaciones requieren de un sistema determinado que ellos escogen según el tipo de licencia.

1.5.3 MySQL como servidor de bases de datos

MySQL es uno de los gestores de base de datos de código abierto más populares actualmente disponibles. Es seguro, rápido y fácil de usar. Fue desarrollado principalmente para manejar grandes bases de datos mucho más rápido que las soluciones existentes. Su conectividad, velocidad y seguridad lo hacen un servidor especialmente apropiado para bases de datos y aplicaciones en Internet. MySQL puede ser integrado tanto al ambiente Windows como Linux, permitiendo la migración y creación de aplicaciones para ambos sistemas operativos. Según estándares de comparación MySQL es más rápido que todos sus coterráneos, incluyendo PostgreSQL. Es el más popular, debido a esto, ha sido más probado y experimentado que los demás, permitiéndole adquirir una mayor estabilidad.

Su instalación es muy fácil y bien documentada. Es compatible con incontables plataformas, incluyendo todas las versiones de Windows. Consume muy pocos recursos, tanto de CPU como de memoria. Posee en su distribución potentes herramientas para chequear sus procesos, sus consultas, para recuperar datos corruptos, etc. Posee un excepcional y granulado control de acceso [MYS08].

1.5.4 Ventajas de MySQL

Ventajas de MySQL:

- ✓ Mayor rendimiento.
- ✓ Mayor velocidad tanto al conectar con el servidor como al ejecutar consultas y demás funciones.
- ✓ Mejores utilidades de administración (backup, recuperación de errores, etc.).
- ✓ Aunque se cuelgue no suele perder información ni corromper los datos.
- ✓ Mejor integración con ZOPE.
- ✓ No hay límites en el tamaño de los registros.

- ✓ Mejor control de acceso, o sea, qué usuarios tienen acceso a qué tablas y con qué permisos.
- ✓ Se comporta mejor que otros gestores de base de datos a la hora de modificar o añadir campos de una tabla.
- ✓

1.5.5 Implementación de los agentes inteligentes

Es una buena opción ligar Zope con lenguaje PHP para la implementación de los entrenadores, pues Zope tiene un producto llamado PHPParse que es un objeto contenedor de código PHP.

Ventajas de usar PHP:

1. Es fácil de aprender comparado con otros lenguajes de programación. El lenguaje es semejante a C y Java, su sintaxis primaria esta basada en Perl. Otro punto es que PHP tiene librerías especializadas en determinados trabajos por lo cual solo se necesita conocer la sintaxis, aplicarla y se logran grandes resultados.
2. Está disponible para la mayoría de sistemas operativos existentes. Desde Linux, Microsoft Windows, MAC, entre otros. Una vez desarrollado tu aplicación PHP esta puede funcionar cualquiera de estos sistemas operativos sin necesidad de modificar el código.
3. Fue diseñado para trabajar sobre la Web por ello trae un conjunto muy amplio de funciones para ser utilizadas en diferentes tareas relacionadas con la Web.
4. Es *Open Source*, es decir, que se tiene acceso al código fuente. Este punto es importante también pues teniendo acceso al código miles de desarrolladores detectan *bugs* y van corrigiendo y mejorando PHP, logrando tener una aplicación muy segura y constantemente mejorada.
5. El precio para utilizar PHP es cero, PHP es gratuito.
6. Tiene una comunidad muy grande de desarrolladores, existen muchos lugares donde se pueden encontrar: documentación, tutoriales, ejemplos de código,

foros, etc. Si se tiene un problema con PHP se puede encontrar la respuesta en muchos sitios en donde los usuarios comparten el conocimiento adquirido en el proceso de desarrollo.

1.6 Concepción general de APA-Prolog

El Ambiente Inteligente de enseñanza-aprendizaje asistido por computadora para la Programación Lógica (APA-Prolog) se desarrolla en el Centro Universitario de Sancti Spiritus con el apoyo de la Universidad Central de Las Villas y la Universidad de Granma.

La confección de APA-Prolog se llevó a cabo en tres etapas fundamentales: Primero la creación del modelo del estudiante, luego la creación del modelo del dominio y finalmente el desarrollo del modelo de adaptación que es el objetivo de esta investigación.

Dicho de esta manera podría pensarse que los modelos anteriores son independientes unos de otros sin embargo es la relación entre ellos y de manera especial con la interfaz de usuario, la que da fortaleza a este tipo de sistemas.

APA-Prolog es, desde el punto de vista práctico, un conjunto de mapas conceptuales relacionados entre sí sobre el cual los estudiantes pueden navegar, encontrando en su travesía recursos informáticos tales como, textos explicativos, simuladores, tutoriales, entrenadores, evaluadores, un directorio temático y un espacio para el intercambio.

Esta navegación, hasta el comienzo de este trabajo, es libre. Es decir, el ambiente no es capaz de tomar en cuenta los conocimientos previos del estudiante.

El Modelo del Estudiante

Para ofrecer un servicio adecuado a cada usuario es necesario que el sistema cuente con una representación de sus características propias, y en base a dicha representación, tome las decisiones pertinentes en la interacción. Para llevar a cabo las tareas anteriores, el sistema utiliza el Modelo del Estudiante.

En este caso el Modelo del Estudiante se obtiene a partir de un diagnóstico inicial, donde el aprendiz se autoevalúa (bien, regular o mal) en cada uno de los conceptos trabajados en el ambiente.

El Modelo del Estudiante de APA-Prolog se caracteriza por:

1. No es un modelo estereotipado sino individualizado para cada aprendiz.
2. Para su creación usa información explícita, ofrecida por el estudiante a través de formularios. Para su actualización incorpora, a la fuente anterior, información obtenida por el sistema a partir de la interacción entre ambos.
3. Es un modelo dinámico y abierto.

Dinámico porque puede cambiar a partir de la actuación del estudiante o a partir de su propio metaconocimiento y abierto porque no está restringido al uso del profesor o al sistema solamente, sino que se expone al usuario para que esté conciente de: ¿Cómo es visto por el sistema? y pueda reflexionar y tomar decisiones que lo animen a mejorar su desempeño [RIO10].

Modelo del dominio

Este modelo es el que contiene toda la información que se desea transmitir a los aprendices, organizada y distribuida para hacerla llegar de la forma más amena, interactiva y motivadora posible.

Se determinó que los mapas conceptuales son una excelente forma de representar el conocimiento por la fuerza expresiva que los caracteriza y la facilidad con que los observadores captan el significado de lo que en ellos se está representando. Es por eso que se emplearon para tejer una gran red con la mayoría de los conceptos correspondientes al lenguaje Prolog, sobre la cual los estudiantes pueden navegar sin correr el riesgo de perderse y no llegar nunca a la información requerida.

Por otra parte, esta representación de la información ha permitido, que desde un inicio, los estudiantes no perciban los elementos del lenguaje de forma aislada, sino relacionados unos con otros.

Hasta donde hemos descrito el conocimiento se encuentra representado en un mapa conceptual plano, por decirlo de alguna manera, donde el estudiante puede apreciar que existe relación entre los conceptos pero no puede profundizar en qué significa

determinado concepto, no puede ver ejemplos, etc. Para resolver esta nueva situación el modelo del dominio ha distribuido el resto de la información en otros recursos informáticos, ellos son: textos explicativos, simuladores [MAR07] [GON07] [MOU07], entrenadores, evaluadores, un espacio para el intercambio [ROD07] y un directorio temático [ROD07]. Cada uno de ellos responde de cierta manera a alguna de las siguientes preguntas: ¿Qué debo saber antes de enfrentarme a este contenido?, ¿Qué aprenderé?, ¿Para qué me sirve?, ¿Cuál es el contenido del material?, ¿Lo he aprendido? y ¿Cómo consolidar lo aprendido?

Una característica común a todos los mapas de APA-Prolog es su jerarquía, esto no es casual, los mapas conceptuales jerárquicos permiten hacer análisis en dos sentidos: de lo general a lo particular y de lo particular a lo general. Por otra parte facilitan la inclusión de nuevos conceptos por parte de los estudiantes, con lo que pueden lograrse buenas sesiones de trabajo colaborativo.

Modelo de adaptación.

El modelo de adaptación es una pieza importante cuando los desarrolladores se trazan la meta de que los sistemas se adapten a las personas. En el siguiente capítulo se describe este modelo en el caso particular de APA-Prolog [RIO10].

1.7 Conclusiones parciales

Los agentes inteligentes constituyen una alternativa para lograr ambientes de enseñanza-aprendizaje con posibilidades de adaptarse a las características de los estudiantes. Zope, MY SQL y PHP son herramientas que facilitan y garantizan una implementación adecuada de sistemas para la enseñanza usando técnicas de IA como son los AI.

CAPITULO 2. DISEÑO E IMPLEMENTACIÓN DEL SISTEMA Y LOS AGENTES INTELIGENTES PARA APA-PROLOG

Los mapas conceptuales se pueden usar para organizar el conocimiento de una manera adecuada gracias a su fuerza expresiva y a la facilidad con que los observadores captan el significado de lo que en ellos está representado. Basado en esas propiedades, la interfaz de APA-Prolog es un mapa conceptual que contiene la mayoría de los conceptos asociados a la Programación Lógica. Esta forma de organizar la información permite relacionar los conceptos y ofrece un medio cómodo de navegación.

La interfaz ofrece una vía de acceso a diferentes recursos informáticos, tales como: textos explicativos, simuladores, entrenadores, evaluadores, un espacio para el intercambio y un directorio temático, los cuales permiten profundizar teóricamente acerca de los conceptos, analizar ejemplos, comprender funcionamientos, intercambiar ideas, etc.

Con el objetivo de proporcionar adaptabilidad a APA-Prolog se diseñó un conjunto de Agentes que se describen a continuación. Se plantea además el análisis del sistema, utilizando para su modelado el Lenguaje Unificado de Modelación (UML), que permite representar el diagrama de casos de uso del sistema y la especificación de los mismos.

2.1 Descripción del sistema propuesto

El trabajo pretende extender el ambiente APA-Prolog introduciendo un conjunto de agentes inteligentes para el trabajo con la Programación Lógica. Los agentes ayudan al estudiante en la adquisición del conocimiento mediante la solución de ejercicios, teniendo en cuenta las condiciones de cada educando. Se requiere también la construcción de una herramienta para la realización de las tareas administrativas y de

actualización del sitio, así como de una aplicación para trabajar con el diagnóstico de los educandos en los diferentes conceptos.

2.1.1 Actores del sistema

Un actor no es parte del sistema, es un rol de un usuario que puede intercambiar información o puede ser un recipiente pasivo de información y representa a un ser humano, a un software o a una máquina que interactúa con el sistema. A continuación se definen los actores del sistema propuesto:

Tabla 2-1: Descripción de los actores del sistema.

| Actores | Justificación |
|-----------------------|--|
| Invitado diagnóstico. | Realiza su diagnóstico desde la aplicación encargada de esta función. |
| Usuario diagnóstico. | Actualiza su diagnóstico desde la aplicación encargada de esta función. |
| Invitado APA-Prolog | Consulta recursos en el mapa conceptual. Revisa bibliografía, realiza ejercicios, trabaja con los evaluadores y los simuladores. |
| Usuario APA-Prolog | Consulta recursos en el mapa conceptual según su estado de conocimiento en los diferentes conceptos de la programación lógica. Revisa bibliografía, realiza ejercicios, trabaja con los evaluadores y los simuladores. |
| Publicador | Encargado de la gestión del contenido referente a los ejercicios desde la herramienta de administración. |
| Administrador | Hace todo lo que el publicador realiza más todo lo que se refiere a la gestión de usuarios y la delegación de permisos. |

2.1.2 Diagrama de casos de usos del sistema

Para una mejor comprensión se decide realizar el diagrama de casos de uso definiendo paquetes. Se muestra un diagrama por cada paquete. Los paquetes de casos de uso son la forma de agrupar a estos últimos respondiendo a algún criterio. Se realizaron tres

paquetes: Administración, Diagnóstico y APA-Prolog. El paquete APA-Prolog depende de los paquetes Administración y Diagnóstico.

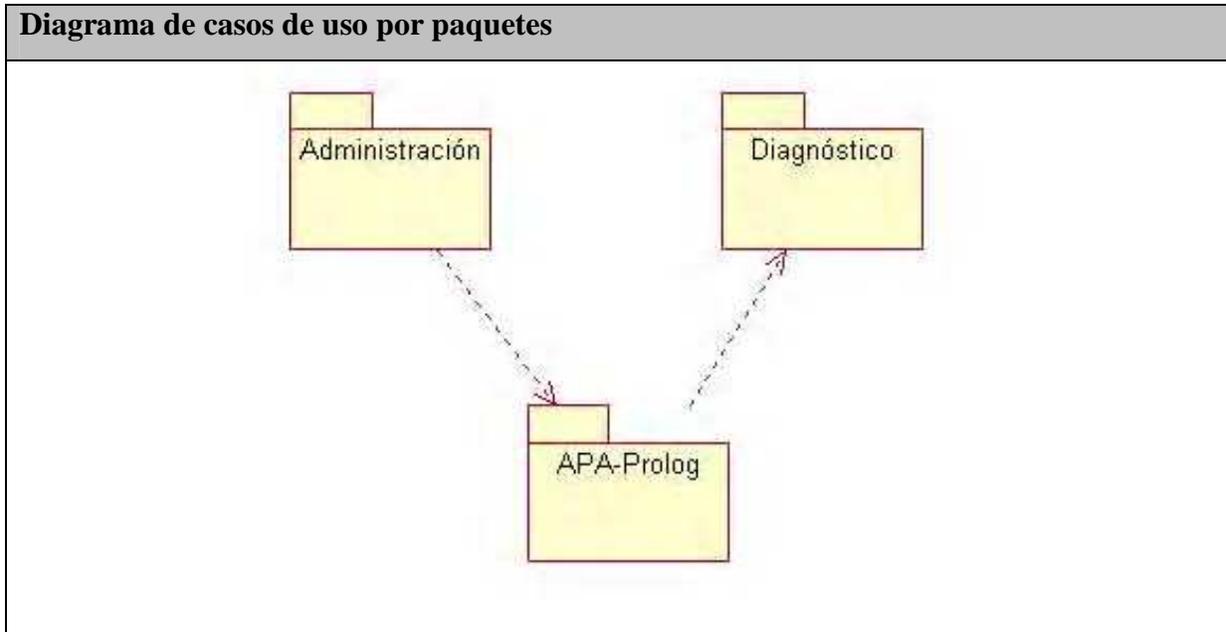


Fig. 2.1: Diagrama de casos de uso por paquetes.

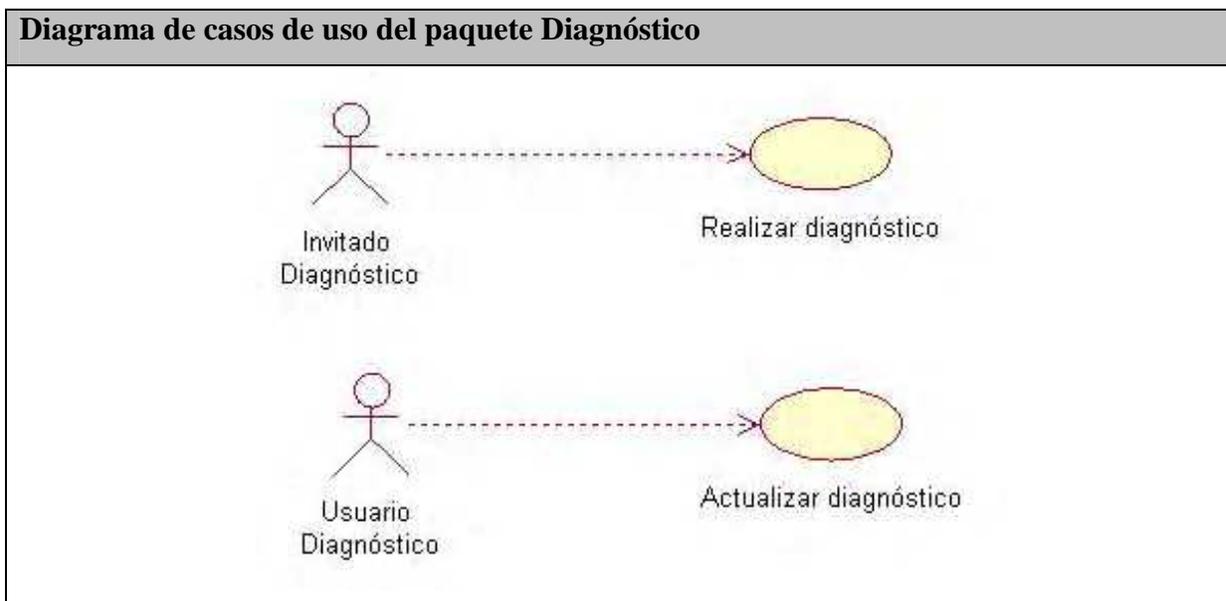


Fig. 2.2: Diagrama de casos de uso del paquete Diagnóstico.

Este paquete recoge los casos de uso para realizar y actualizar el diagnóstico. Estos datos son imprescindibles para realizar las diferentes tareas teniendo en cuenta el estado de conocimiento de los estudiantes en los diferentes conceptos.

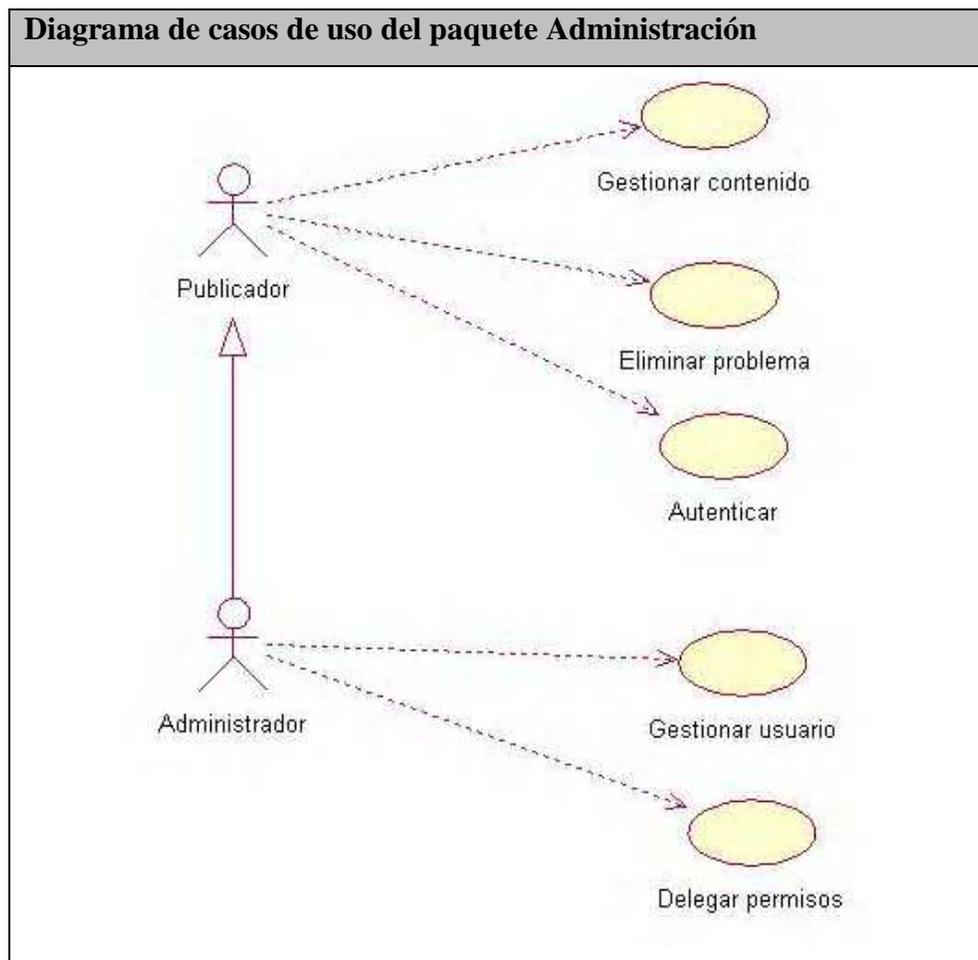


Fig. 2.3: Diagrama de casos de uso del paquete Administración.

Este paquete recoge los casos de uso para realizar las tareas de administración y gestionar la información referente a los diferentes problemas a los que se enfrentarán los estudiantes.

Diagrama de casos de uso del paquete APA-Prolog

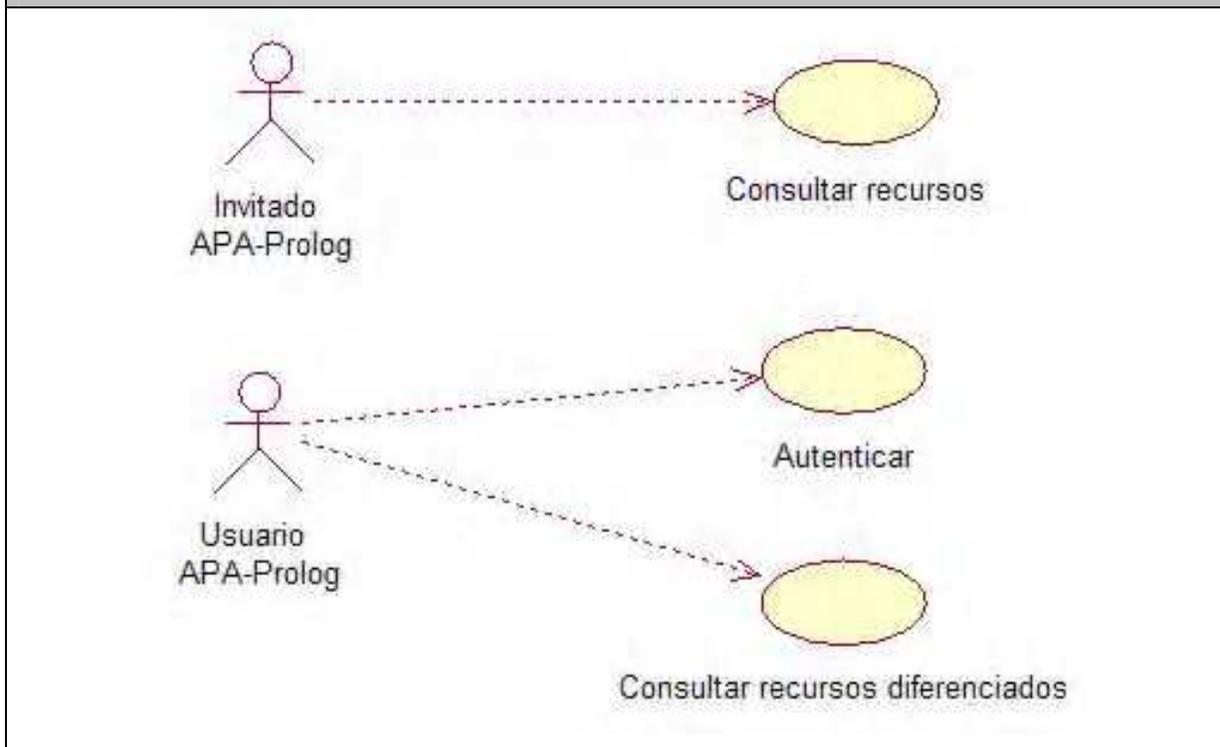


Fig. 2.4: Diagrama de casos de uso del paquete APA-Prolog.

Este paquete recoge los casos de uso referentes a las consultas de recursos del mapa conceptual y la autenticación en la aplicación APA-Prolog.

2.1.3 Descripción de los casos de usos del sistema

A través de la tabla 2.2 se enuncian y describen los casos de uso del sistema.

Tabla 2-2: Descripción de los casos de uso del sistema.

| | |
|-------------------|--|
| CU-1 | Realizar diagnóstico |
| Actores: | Invitado diagnóstico |
| Propósito: | Realizar un diagnóstico al estudiante. |
| Resumen: | Una vez que el invitado hace su diagnóstico se le crea una cuenta para autenticarse en la aplicación APA-Prolog. Este diagnóstico se tendrá en cuenta para mostrar el contenido según el estado de conocimiento del estudiante en un concepto determinado. |

| | |
|------------------------|--|
| Poscondiciones: | Se le crea una cuenta al usuario y queda bien concebido su estado de conocimiento en los diferentes conceptos. |
|------------------------|--|

| | |
|------------------------|--|
| CU-2 | <i>Actualizar diagnóstico</i> |
| Actores: | Usuario diagnóstico. |
| Propósito: | Actualizar el diagnóstico del estudiante. |
| Resumen: | El estudiante actualiza su diagnóstico y con ello su estado de conocimiento. |
| Poscondiciones: | El estado de conocimiento del estudiante queda actualizado y a partir de este momento se tendrá en cuenta esta modificación para mostrar los recursos en APA-Prolog. |

| | |
|------------------------|---|
| CU-3 | <i>Gestionar contenido</i> |
| Actores: | Publicador y Administrador |
| Propósito: | Gestionar todo el contenido referente a los problemas. |
| Resumen: | Adicionar y actualizar problemas, incisos y respuesta a incisos. |
| Poscondiciones: | Los problemas quedan listos para mostrarse más tarde en los entrenadores y los evaluadores. |

| | |
|-------------------|---|
| CU-4 | <i>Eliminar problema</i> |
| Actores: | Publicador y Administrador |
| Propósito: | Eliminar problema. |
| Resumen: | Eliminar problema completo, un inciso o una respuesta a inciso. |

| | |
|------------------------|--|
| CU-5 | <i>Autenticar</i> |
| Actores: | Administrador |
| Propósito: | Autenticarse en el sistema de administración. |
| Resumen | El administrador entra su usuario y contraseña para entrar al sistema. |
| Poscondiciones: | Todo queda listo para que el administrador realice determinadas tareas orientadas al manejo de usuario o la manipulación de los problemas. |

| | |
|------------------------|--|
| CU-6 | <i>Gestionar usuario</i> |
| Actores: | Administrador |
| Propósito: | Gestionar los usuarios en el sistema. |
| Resumen: | Adicionar, actualizar y eliminar información referente a los usuarios. |
| Precondiciones: | El publicador previamente debe autenticarse. |

| | |
|------------------------|---|
| CU-7 | <i>Delegar permisos</i> |
| Actores: | Administrador |
| Propósito: | Establecer el nivel de acceso de los usuarios. |
| Resumen: | Crear diferentes roles y asignar un rol determinado a los usuarios. |
| Precondiciones | El administrador previamente debe autenticarse. |
| Poscondiciones: | Queda establecido el nivel de acceso en el sistema. |

| | |
|-------------------|--|
| CU-8 | <i>Consultar recursos</i> |
| Actores: | Invitado APA-Prolog |
| Propósito: | Consultar los diferentes recursos disponibles en APA-Prolog, |
| Resumen: | Se consulta los diferentes recursos sin tener en cuenta el estado de conocimiento del invitado en los diferentes conceptos que se manejan en la aplicación. Estos recursos pueden ser: simuladores, bibliografía, entrenadores, evaluadores etc. |

| | |
|-----------------------|---|
| CU-9 | <i>Autenticar</i> |
| Actores: | Usuario de APA-Prolog |
| Propósito: | Autenticarse en APA-Prolog. |
| Resumen | El usuario entra su usuario y contraseña para entrar al sistema. |
| Poscondiciones | Se le muestran los recursos al usuario según su estado de conocimiento en los diferentes conceptos. |

| | |
|------------------------|---|
| CU-10 | <i>Consultar recursos</i> |
| Actores: | Usuario APA-Prolog |
| Propósito: | Consultar los diferentes recursos disponibles en APA-Prolog, |
| Resumen: | Se consulta los diferentes recursos teniendo en cuenta el estado de conocimiento del invitado en los diferentes conceptos que se manejan en la aplicación. Los recursos se muestran según la evaluación que posea el usuario (bien, regular, mal). Estos recursos pueden ser: simuladores, bibliografía, entrenadores, evaluadores etc. |
| Precondiciones: | El usuario primeramente realiza su diagnóstico y luego se autentica en APA-Prolog. |
| Poscondiciones | En caso de que consulte algún evaluador y tiene resultados satisfactorios, se actualizará su estado de conocimiento en el concepto correspondiente. |

| | |
|------------------------|--|
| CU-11 | <i>Autenticar</i> |
| Actores: | Publicador |
| Propósito: | Autenticarse en el sistema de administración. |
| Resumen | El publicador entra su usuario y contraseña para entrar al sistema. |
| Poscondiciones: | Todo queda listo para que el publicador realice determinadas tareas orientadas a la manipulación de los problemas. |

2.1.4 Diseño de la base de datos

Nuestro sistema cuenta con una base de datos implementada en MySQL versión 5.0.21-nt. En esta base de datos se almacena toda la información referente a los problemas, los incisos, la respuesta a los incisos, la categoría de la respuesta, dificultad de los ejercicios, tipo de recurso al que pertenece el problema y el por ciento que se tiene en cuenta a la hora de realizar la evaluación de un ejercicio. La base de datos cuenta con siete tablas donde se almacena toda la información necesaria (ver Fig. 2.5).

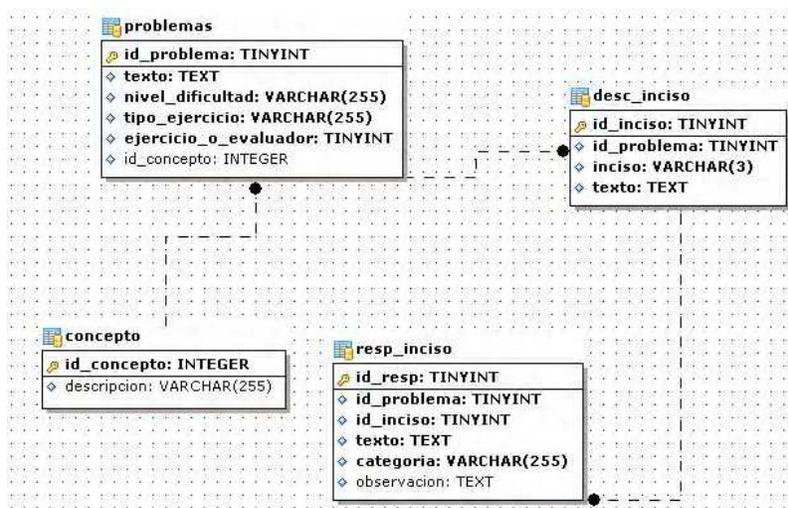


Fig. 2.5: Diseño de la base de datos.

Tabla 2-3: Descripción de las tablas de la base de datos.

| Tabla | Descripción |
|-------------|---|
| problemas | Contiene los problemas |
| concepto | Contiene los diferentes conceptos de la Programación Lógica |
| desc_inciso | Contiene los incisos |
| resp_inciso | Contiene la información de la respuesta de los incisos |

2.1.5 Diagramas de navegación

En nuestro sistema cada estado está representado por una página que conforma una vista en el navegador. El diagrama de navegación está formado por un conjunto de estados y sus correspondientes enlaces, a través de los cuales un usuario puede transitar de un estado a otro con la posibilidad de retornar a la página inicial cuando se estime conveniente.

A continuación se definirán los estados a través de los cuales transita un usuario (autenticado en el mapa conceptual, publicador y administrador) partiendo de su respectivo estado inicial.

Diagrama de navegación para usuario autenticado en el mapa conceptual:

Un usuario autenticado en el mapa conceptual, a partir del estado inicial, (página principal) puede acceder a las diferentes opciones que brinda el sistema (ver Fig. 2.6).

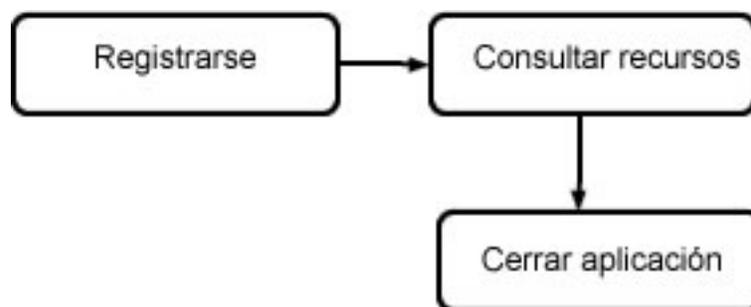


Fig. 2.6: Diagrama de navegación del usuario autenticado en el mapa conceptual.

Diagrama de navegación del usuario publicador:

El usuario publicador tendrá como estado inicial una sección de autenticación para acceder a otros estados dirigidos a tareas de actualización, entre las que se encuentran: edición de problemas, incisos, respuesta a incisos y por ciento (ver Fig. 2.7).

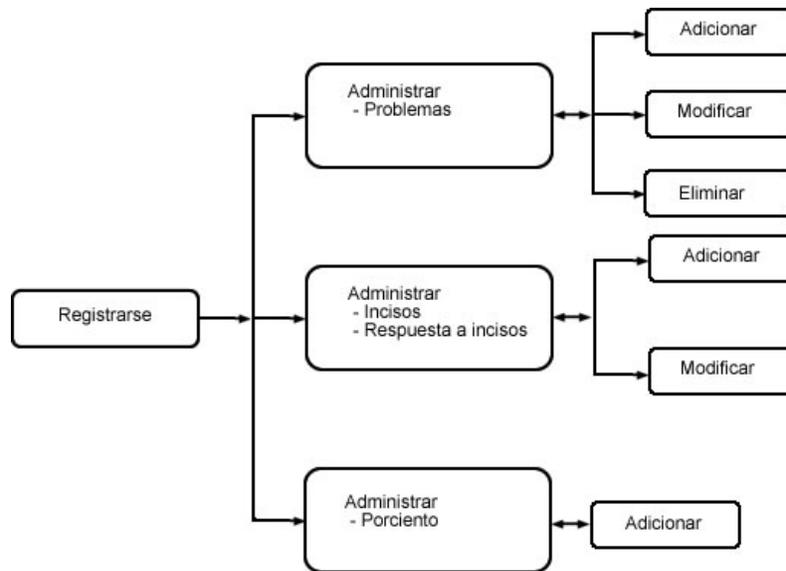


Fig. 2.7: Diagrama de navegación del usuario publicador.

Diagrama de navegación del usuario administrador:

El usuario administrador tendrá como estado inicial una sección de autenticación para acceder a otros estados dirigidos a tareas administrativas, entre las que se encuentran: crear, eliminar y actualizar usuarios, así como delegar permisos (ver Fig. 2.8).

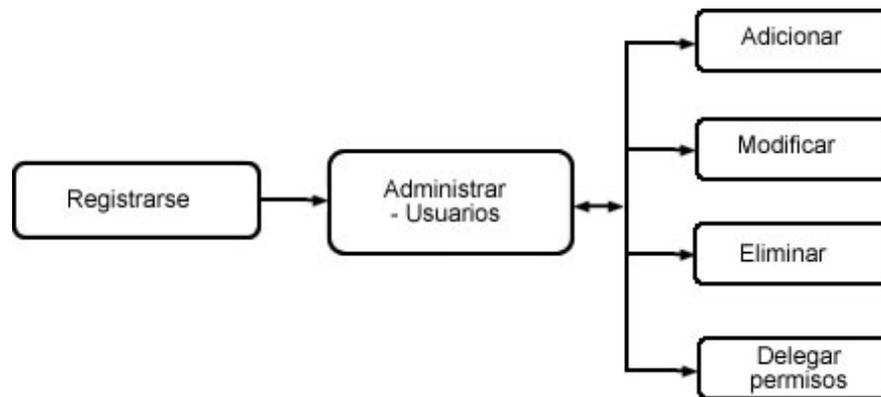


Fig. 2.8: Diagrama de navegación del usuario administrador.

2.2 Arquitectura básica de un agente

Cada uno de los agentes de APA-Prolog tiene una tarea específica que cumplir. Aplicando la técnica “Divide y Vencerás” cada una de ellas da lugar a otro grupo de actividades que los agentes deben realizar para conseguir un resultado exitoso. Estas actividades, de forma general, originan un conjunto de reglas que relacionan el estado de conocimiento del estudiante acerca del concepto estudiado con los recursos informáticos disponibles. La figura 2.9 muestra la arquitectura básica de un agente. Seguidamente se describen las características individuales de cada uno de ellos.

2.2.1 Agente Adis

La necesidad de crear ambientes computacionales capaces de mantener el interés de sus usuarios, implica el desarrollo de interfaces personalizadas para ofrecer un servicio que permita un trato individualizado al usuario en particular [MED07], adaptando la interacción con el sistema a necesidades e intereses personales. Para ofrecer un servicio adecuado a cada usuario es necesario que el sistema cuente con una representación de sus características propias, y en base a dicha representación, tome las decisiones pertinentes en la interacción [FIS00]. Para llevar a cabo las tareas anteriores, el sistema realiza un diagnóstico inicial a cada uno de los estudiantes que interactúan con el ambiente siendo esta la misión asignada al agente Adis.

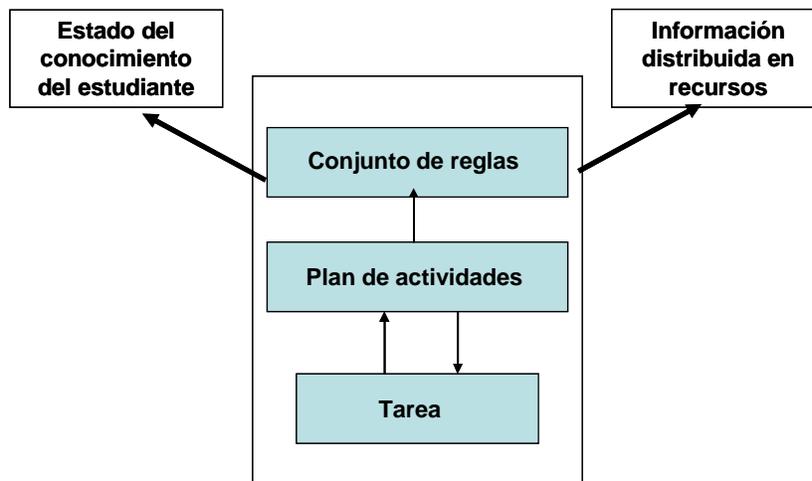


Fig. 2.9 Arquitectura básica de un agente de APA-Prolog

Adis usa información explícita inicial que solicita al estudiante a través de formularios y la almacena en una base de datos relacional. Posteriormente, la información puede actualizarse a través de dos vías: la primera, a partir de la interacción entre el sistema y el usuario y la segunda, de la misma forma que la inicial.

Los datos que maneja Adis se concibieron abiertos y dinámicos. Dinámicos porque pueden cambiar a partir de la actuación del estudiante o a partir de su propio metaconocimiento y abiertos porque no está restringido al uso del profesor o al sistema solamente sino que se expone al usuario para que esté consciente de cómo es visto por el sistema y pueda reflexionar y tomar decisiones que lo animen a mejorar su desempeño.

Para lograr todo lo anterior Adis desarrolla las siguientes actividades:

1. Apropiarse del nombre y la palabra clave que usará el estudiante para autenticarse.
2. Capturar el grado de conocimiento que posee el estudiante sobre cada uno de los conceptos fundamentales del ambiente y recordarlos (véase la figura 2.10).
3. Investigar si el estudiante tiene algún objetivo específico que vencer, por ejemplo: los hechos, las reglas, las preguntas, etc.
4. Almacenar esos objetivos específicos.

5. Facilitar al aprendiz los elementos necesarios para autoevaluarse.

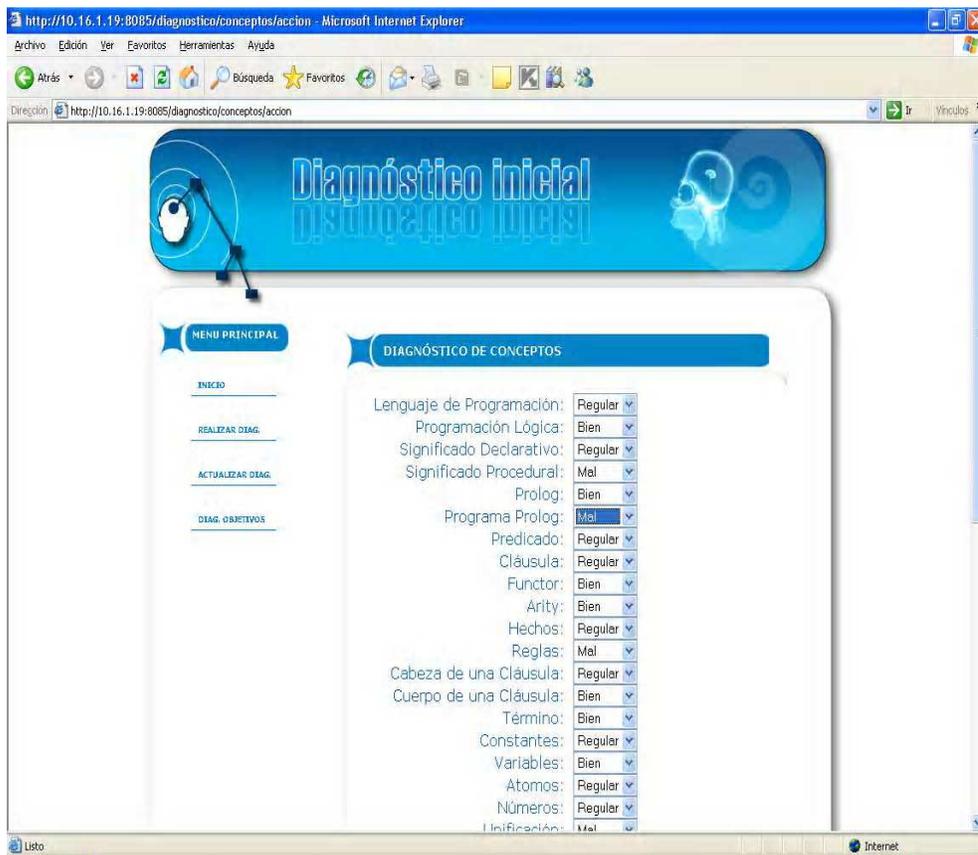


Fig. 2.10 Formulario que permite al estudiante autodiagnosticarse

Esta última actividad merece un comentario adicional, pues generalmente los estudiantes no saben medir sus conocimientos sobre algunos temas y responden bien, regular o mal teniendo en cuenta sólo lo que han leído o han escuchado al respecto. Sin embargo, Adis pretende actuar en el campo del metaconocimiento del estudiante, diciéndole por ejemplo: “... para tener un grado de conocimiento bueno acerca de las reglas usted debe saber, ante un problema dado, escribir las reglas correspondientes; si no sabe hacer esto pero es capaz de interpretar las ya existentes, entonces su conocimiento es regular y si no puede hacer ninguna de las dos entonces es mal”. La figura 2.11 muestra la arquitectura de Adis, que constituye un caso particular de la arquitectura básica pues este agente no contiene reglas ni se relaciona con la

información a mostrar. Para hacer esta información perdurable se implementó una base de datos relacional en MySQL.

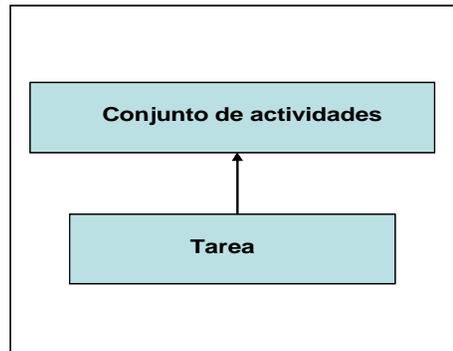


Fig. 2.11 Arquitectura de Adis

La base de datos la usan, como fuente de información, los agentes Tivo, Teo, Tica y Eva para conocer el estado de conocimiento del estudiante con que están trabajando. Eva también tiene autoridad para actualizarla cuando el aprendiz haya obtenido resultados que impliquen cambios en su evaluación y pueda, por tanto, tener un estado de conocimiento diferente. La figura 2.12 ilustra esa relación. En el epígrafe 2.5 se expone cómo se logra este intercambio a través del modelo del pizarrón.

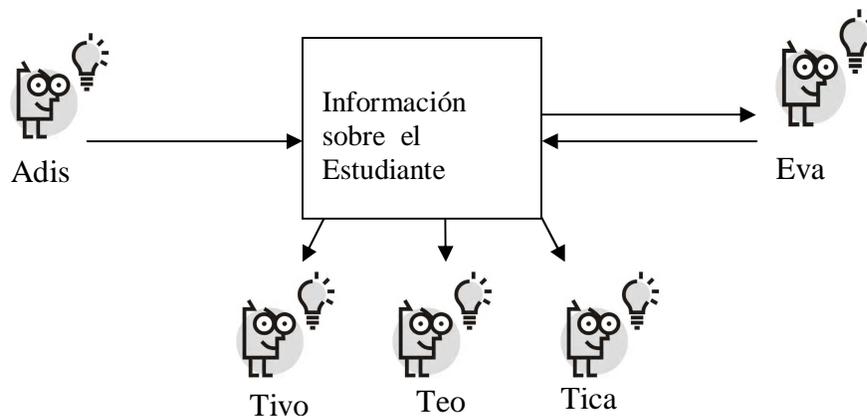


Fig. 2.12 Interacción de los A.I con la información acerca del estudiante

Se decidió usar MySQL porque es uno de los gestores de base de datos de código abierto más populares disponibles actualmente. Es seguro, rápido y fácil de usar. Fue desarrollado principalmente para manejar grandes bases de datos mucho más rápido que las soluciones existentes. Su conectividad, velocidad y seguridad lo hacen un servidor especialmente apropiado para bases de datos y aplicaciones en Internet, la aplicación que se construye actúa a distancia y por tanto se precisa que la demora no sea significativa. MySQL puede integrarse tanto al ambiente Windows como Linux, permitiendo la migración y creación de aplicaciones para ambos sistemas operativos. Su instalación es muy fácil y bien documentada. Es compatible con múltiples plataformas, incluyendo todas las versiones de Windows. Consume muy pocos recursos, tanto de CPU como de memoria. Posee en su distribución herramientas para chequear sus procesos, sus consultas y para recuperar datos corruptos, entre otras.

2.2.2 Agente Tivo

El agente Tivo tiene por objetivo facilitar el camino hacia informaciones específicas basadas en insuficiencias puntuales, para las cuales recomienda los caminos y recursos a utilizar, por ejemplo, si se conoce que un estudiante tiene dificultades en el concepto de lista, Tivo podría dar la siguiente recomendación: "... debes revisar el mapa llamado Listas, consultar el simulador de Listas y los materiales M1, M2 y M3...".

El agente no ayuda al estudiante a construir su conocimiento, sólo le facilita los materiales necesarios para ello, acortándole el tiempo de navegación y evitando que se pierda y deje de utilizar algunos recursos que pudieran aportar conocimientos significativos.

Tivo proporciona una navegación adaptativa para lo cual desempeña tres actividades fundamentales:

1. Propone una lista, ordenada por relevancia, de todos los materiales disponibles en APA-Prolog que pudiera ayudar a las necesidades cognitivas del estudiante.

2. Muestra la secuencia de pasos para lograr el objetivo actual, sugiriendo los enlaces más importantes a seguir, por ejemplo, si el objetivo son los predicados, el camino indicado puede ser: Prolog → Programas → Predicado.
3. Alerta acerca de los conceptos que son precedentes al concepto objetivo, por ejemplo, si el objetivo fuera predicados, los prerrequisitos serían los conocimientos relacionados con las cláusulas, el functor y el arity de las cláusulas, entre otros.

2.2.3 Agente Teo

Los elementos teóricos son de suma importancia en cualquier proceso de enseñanza-aprendizaje. El agente Teo es capaz de decidir qué materiales teóricos, de todos los que hay disponibles en el ambiente, son los adecuados para un estudiante en particular. Para ello se apoya en la representación del conocimiento del estudiante y realiza las siguientes acciones:

1. Usando las técnicas de mapas adaptativos, y ocultamiento adaptativo de enlaces determina qué variante de mapas mostrar, adicionando o quitando enlaces para mostrar una configuración adaptada a su nivel.
2. Decide qué textos presentar en cada concepto, implementando para eso las siguientes técnicas:
 - a) Variantes de páginas: El sistema almacena dos o tres versiones de una misma página, estas se diferencian en la cantidad de información que contienen y en la profundidad con que es tratada.
 - b) Explicación de variantes: El contenido tiene diferentes versiones de explicaciones que se muestran de acuerdo al estado de conocimiento del estudiante.
 - c) Coloca en orden descendente de relevancia los enlaces a la información más adecuada según los conocimientos previos del estudiante.

3. Determina los ejemplos a visualizar, valiéndose de las técnicas anteriores.
4. Suministra una lista ordenada de bibliografía recomendada.
5. Sugiere el orden del próximo concepto a visitar desde el nodo actual.
6. Inserta definiciones de todos los conceptos presentes entre los prerrequisitos para cada nodo.
7. Recomienda ejercicios para entrenamiento y en ocasiones el uso de entrenadores.
8. Recomienda ver simuladores.
9. Recomienda evaluaciones.
10. Motiva para el próximo concepto a visitar.

Para desarrollar con éxito la tarea asignada, Teo manipula diferentes recursos, basándose en un conjunto de reglas que tienen el formato general siguiente:

```
Si estado de conocimiento (C) = <evaluación>
    entonces mostrar <medio>
```

donde:

C representa el concepto que se está tratando,

<evaluación> toma valores del conjunto {bien, regular, mal},

<medio> representa un subconjunto (diferente del vacío) de los recursos disponibles para Teo (mapas conceptuales, textos explicativos, simulaciones, presentaciones, tutoriales)

2.2.4 Agente Tica

Tica es el agente encargado de presentar las actividades prácticas que debe realizar el estudiante, las cuales están asociadas a su nivel de conocimiento. El agente realiza las siguientes actividades:

Presenta los ejercicios a resolver y las guías para su solución.

Ofrece enlaces a prerrequisitos.

Refuerza las respuestas correctas.

No penaliza las respuestas incorrectas sino que proporciona elementos importantes a tener en cuenta para llegar a la solución correcta (explicaciones adicionales, explicaciones de prerrequisitos, explicaciones de variantes y explicaciones comparativas).

Si el estudiante no llega a la solución correcta, entonces recomienda estudiar la teoría necesaria para conseguirlo (explicaciones adicionales y variante de páginas).

Si el estudiante responde acertadamente el cuestionario se le recomiendan documentos y materiales para profundizar.

Recomienda pasar al evaluador.

Los entrenadores son los recursos con que cuenta Tica para realizar sus actividades, ellos están asociados a aquellos conceptos que requieren ser ejercitados.

2.2.4.1 Entrenadores

En ningún ambiente para el aprendizaje, real o virtual deben faltar las actividades prácticas. En esta ocasión se concibieron mediante entrenadores inteligentes esencialmente orientados al desarrollo de habilidades, pues no pretenden la dirección total del proceso de instrucción ni llevan a cabo la formación de conceptos nuevos. Supervisan la actividad práctica del estudiante mediante el control de los errores durante la solución de los ejercicios, hacen recomendaciones y controlan la presentación dosificada de problemas y ejercicios [MED07].

Teniendo en cuenta la cantidad de entrenadores que pudiera ser necesaria para ambientes de enseñanza-aprendizaje relacionados con la programación y dado el carácter eminentemente práctico que la identifica, se creó una herramienta que puede usarse para desarrollar tantos ejercitadores como se requiera y sin inconvenientes para

extrapolarlos a otras asignaturas u otros entornos de aprendizaje. De modo que puede hablarse de dos aplicaciones:

- Una herramienta de Administración (HerAdE), con la que trabaja el profesor o persona encargada de situar los ejercicios debidamente graduados por complejidad.
- El entrenador, con el que interactúan los estudiantes.

HerAdE tiene un sistema de seguridad que permite mantener el control del personal con acceso a:

- Definir los conceptos con que se va a trabajar.
- Añadir problemas.
- Adicionar incisos a los problemas.
- Agregar respuestas correctas e incorrectas a los incisos del problema.
- Establecer el índice con que se comparará el por ciento de respuestas correctas.
- Modificar o eliminar cualquiera de los elementos anteriores.

Esta información se almacena en una base de datos relacional cuyas tablas fundamentales son:

Concepto (identificador_c, descripción)

Problema (identificador_p, nivel_de_dificultad, concepto_al_que_está_asociado)

Incisos (identificador_i, identificador_p, texto_descriptivo)

Respuestas_incisos (identificador_r, identificador_p, identificador_i, texto, categoría)

Donde:

- categoría puede ser: correcta o incorrecta y
- nivel de dificultad puede tomar alguno de los valores: alto, medio o bajo.

Esta herramienta cuenta con dos niveles de administración:

1. El administrador del Zope que tiene facultad para crear nuevos usuarios con privilegios de administración sobre la herramienta antes descrita.
2. Los otros administradores (publicadores) que son los encargados de entrar toda la información relacionada con los ejercicios.

La herramienta para la ejercitación tiene como objetivo facilitar la realización de actividades prácticas por parte de los estudiantes y servirle de guía durante la sesión de trabajo. Hace un balance de la cantidad de incisos resueltos de modo correcto e incorrecto y ofrece una valoración general de la actuación del estudiante frente al contenido tratado. Del mismo modo procede con la cantidad de problemas que el alumno resuelve correcta e incorrectamente para hacerle las sugerencias y recomendaciones oportunas. Si de modo general las respuestas del estudiante son correctas se le exhorta a pasar al Evaluador (ver figura 2.13).



Fig. 2.13 Vista del entrenador para el concepto Reglas.

2.2.5 Agente Eva

El agente Eva tiene los privilegios de evaluar al estudiante y decidir si modifica o no su estado de conocimiento, en caso afirmativo lo modifica, situándole valores de bien, regular o mal. Se desempeña de la siguiente manera:

1. Selecciona el cuestionario teniendo en cuenta el concepto y el estado de conocimiento del estudiante en ese concepto.
2. Evalúa cualitativamente.
3. Recomienda pasar o no al siguiente concepto en el mapa.
4. Sugiere otros recursos para apropiarse del contenido cuando el estudiante no es capaz de responder correctamente alguna pregunta del examen.
5. Orienta bibliografía para profundizar si el estudiante ha respondido correctamente.
6. No penaliza al estudiante si los resultados no son los esperados sino que hace recomendaciones para superar las dificultades encontradas.
7. Si el estudiante responde con éxito el formulario que le corresponde, pasa su estado de conocimiento a un nivel superior (bien o regular).
8. Si el estudiante no responde con éxito el formulario que le corresponde entonces pasa su estado de conocimiento a un nivel inferior (regular o mal).

2.2.5.1 Evaluadores

En cualquier proceso de enseñanza-aprendizaje se distinguen tres grandes momentos: la introducción del nuevo contenido, la ejercitación y la evaluación. Por tanto, no resulta difícil entender que si se asumen transformaciones en el diseño e implementación de los medios de enseñanza, en las formas de organización, en el sistema de tareas y el trabajo independiente de los estudiantes, también hay que llevar a cabo transformaciones en el sistema de evaluación.

Los evaluadores son los medios con que cuenta Eva para llevar a cabo su trabajo. Un evaluador es un recurso que tiene la finalidad de brindar un cuestionario, a modo de conclusión, de un concepto específico.

Si dicho cuestionario se llena de forma exitosa, significará que el aprendiz ha pasado a un grado de conocimiento superior lo que se refleja automáticamente en el estado de conocimiento del estudiante sobre ese concepto. Los evaluadores se confeccionaron de modo muy similar a los entrenadores, utilizando la misma aplicación.

Con el propósito de propiciar un ambiente de trabajo cómodo, APA-Prolog dispone de otros dos recursos denominados espacio para el intercambio y directorio temático. A continuación se describen los recursos mencionados.

2.3 Modelo para la solución cooperativa

Como ya se ha expuesto, APA-Prolog basa su trabajo en la actuación de cinco Agentes Inteligentes con capacidad para ejecutar diversas técnicas de adaptación. Estos agentes tienen vida propia pero interactúan entre sí para ofrecer contenidos personalizados e influir en la motivación del aprendizaje mediante el ofrecimiento de contenidos de retroalimentación, ejercicios adaptados al nivel de conocimiento y una interfaz atractiva.

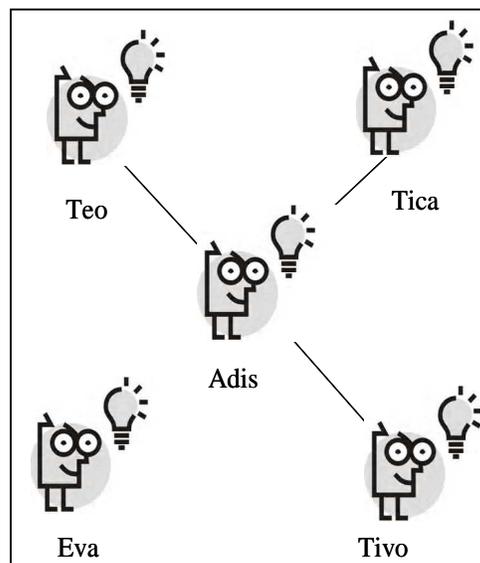


Fig. 2.14 Interacción entre agentes.

El agente Adis obtiene la información primaria acerca del estado de conocimiento del estudiante, esa información la necesitan los demás agentes para realizar su trabajo, de ahí la necesidad de la existencia de relaciones de comunicación entre Adis y los restantes agentes (ver Fig. 2.14). La vía para interactuar con la información se apoya en el modelo del pizarrón (ver Fig. 2.15).

El pizarrón, para el conjunto de agentes que sustentan el trabajo de APA-Prolog, está representado por la clase Estudiante, de modo que, cuando un aprendiz intenta entrar al ambiente lo primero que se hace es verificar si ya se diagnosticó (este hecho implica que ya existe como usuario). En ese caso:

1. Se crea una instancia de la clase Estudiante y se llena con todos los estados de conocimiento del estudiante que se obtienen de una consulta a la base de datos.
2. Se crea una sesión con dos objetivos: el primero, poder llevar la información de la instancia creada de una página Web a la otra de forma que en cada página se pueda preguntar por el estado del conocimiento acerca de un concepto determinado; el segundo, que varios usuarios puedan trabajar a la misma vez en sesiones diferentes.
3. Cuando el estudiante abandona el sistema se actualiza la base de datos a partir de la instancia de la clase y se cierra la respectiva sesión.

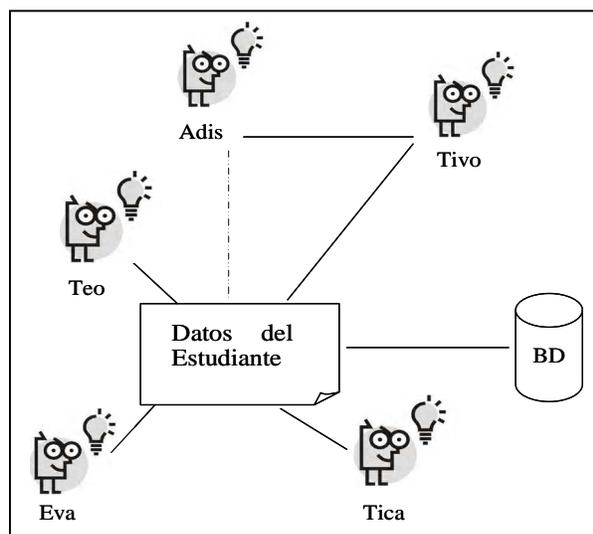


Fig. 2.15 Modelo para la solución cooperativa en APA-Prolog.

De forma que la conducta general del sistema multiagente es como se ilustra en la figura 2.16 y el funcionamiento general del ambiente, una vez diagnosticado el estudiante como se muestra en la figura 2.17.

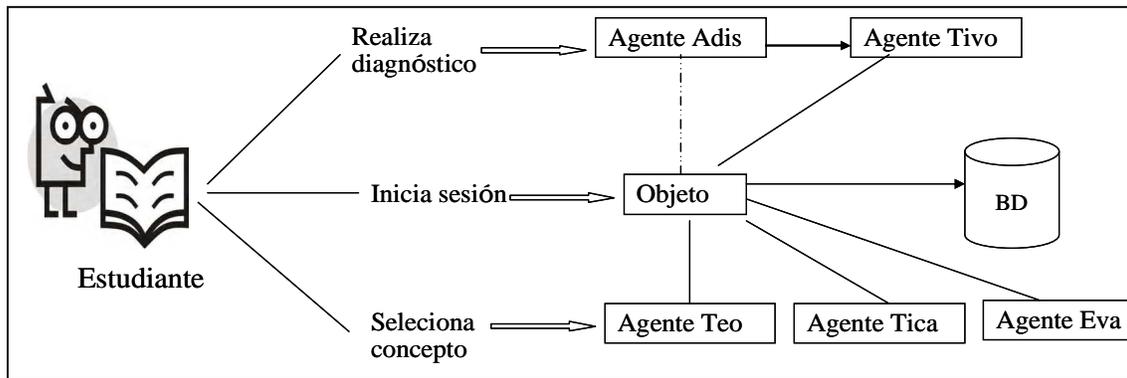


Fig. 2.16 Conducta del sistema multiagente en APA-Prolog.

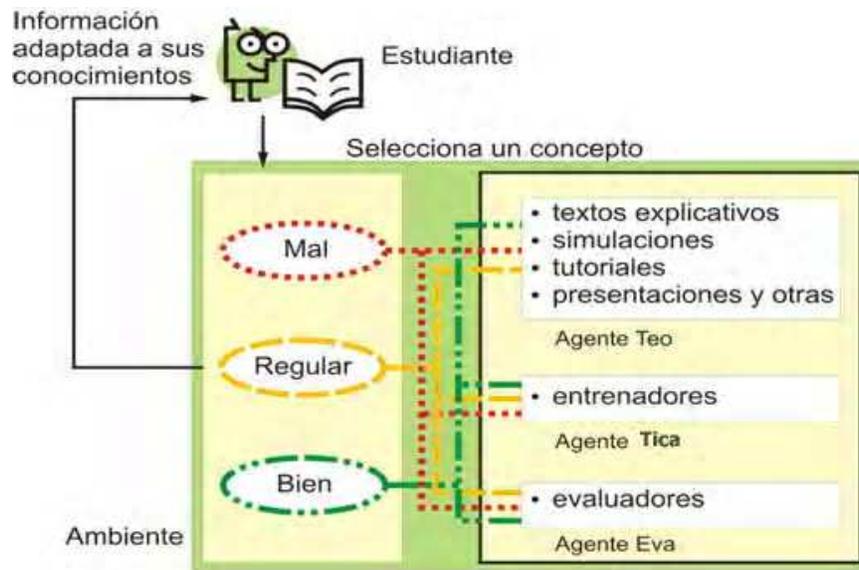


Fig. 2.17 Representación del funcionamiento general del sistema.

Este funcionamiento hace posible que, para un estudiante cuyo estado de conocimiento sobre el concepto Programa Prolog sea regular se muestre un mapa con los recursos

que se aprecian en la figura 2.18 y para un estudiante evaluado de mal en el mismo concepto se muestre un mapa como el de la figura 2.19.

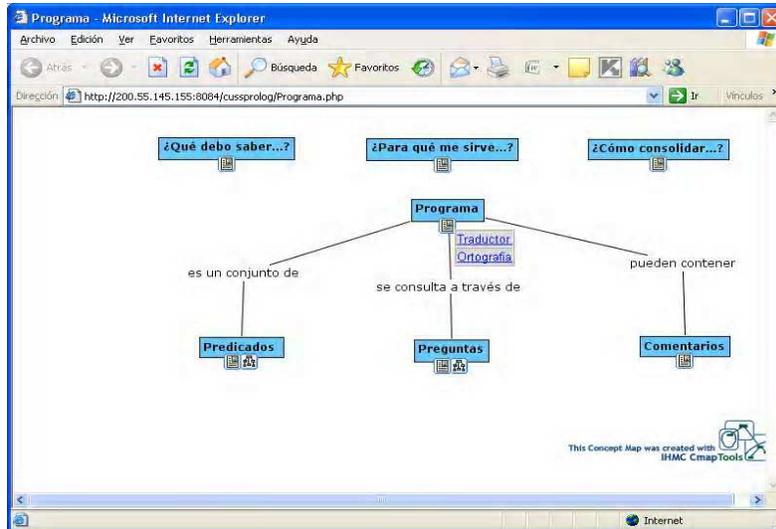


Fig. 2.18 Mapa conceptual para un estudiante.

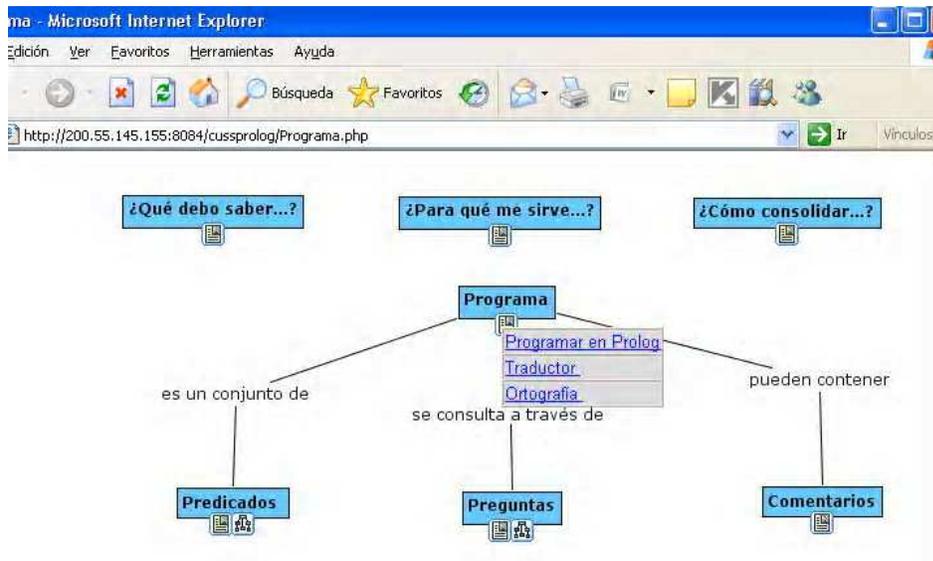


Fig. 2.19 Mapa conceptual para un estudiante evaluado de mal.

2.4 Análisis de los resultados valorados por expertos

Se utilizó el método Delphi para la selección y procesamiento de las encuestas a los expertos. En el Anexo 1 aparecen los aspectos que se tuvieron en cuenta para la selección de los expertos, los que emitieron sus criterios en cuanto al ambiente que se propone.

Para elegir la muestra se tuvo en cuenta a expertos de más de diez años de experiencia como profesionales y profesores de las disciplinas Programación de Computadoras e Inteligencia Artificial, con dominio de la metodología de la enseñanza de la programación y en el diseño de software educativo. De los 35 expertos inicialmente previstos, 30 cumplieron con los requisitos planteados en la encuesta, de ellos el 80.0% con un nivel de competencia Alta y el 20.0% Media, todos tienen categorías docentes principales, 46.7 % son Auxiliares, 53.3% Titulares, 26.7 % son Máster y 73.3% Doctores (Anexo 2).

A los seleccionados se les mostró en detalles el ambiente propuesto, y se les aplicó una encuesta con los elementos a evaluar (Anexo 3).

2.4.1 Procesamientos estadísticos utilizados

En esta parte del trabajo se hace un análisis descriptivo de datos, utilizando en general frecuencias para variables dicotómicas y proporciones de respuestas positivas.

2.4.2 Valoraciones de los expertos

La Tabla 2.4 muestra los resultados de la encuesta a los expertos, sobre la forma en que considera que este ambiente permite elevar los niveles de asimilación de los contenidos relacionados con la Programación Lógica. Puede observarse que el 66,7 % opina que es muy adecuada y el restante 33,3 % que es adecuado.

Tabla 2-4: Respuesta de los expertos a la pregunta 1 del anexo 3.

| Respuestas | Frecuencia | Por ciento |
|--------------|------------|------------|
| Adecuado | 10 | 33,3 |
| Muy adecuado | 20 | 66,7 |
| Total | 30 | 100,0 |



Fig. 2.20 Sobre resultados Tabla 2-4.

La posibilidad que tiene el ambiente de tomar en cuenta los conocimientos previos de los estudiantes fue considerada por el 53,3 % de los expertos como muy buena, mientras que el 46,7 % opina que es buena. (Tabla 2.5)

Tabla 2-5: Respuesta de los expertos a la pregunta 2 del anexo 3.

| | Frecuencia | Por ciento |
|-----------|------------|------------|
| Buena | 14 | 46,7 |
| Muy buena | 16 | 53,3 |
| Total | 30 | 100,0 |

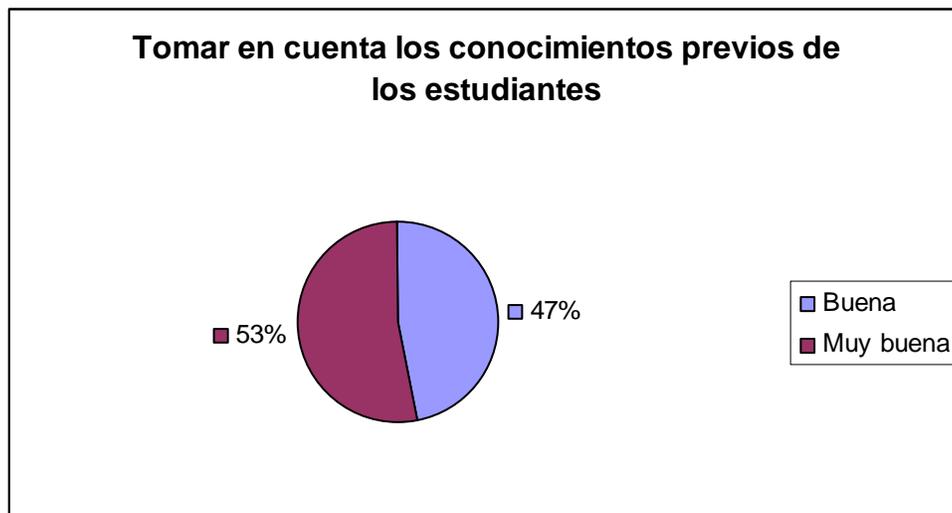


Fig. 2.21 Sobre resultados Tabla 2-5.

Sobre el diagnóstico propuesto como punto de partida para la adaptación, el 86,6 % lo consideró muy apropiado, mientras que el 13,3 % opinó fue apropiado.

Tabla 2-6: Respuesta de los expertos a la pregunta 3 del anexo 3.

| | Frecuencia | Por ciento |
|---------------|------------|------------|
| Muy apropiado | 26 | 86,6 |
| Apropiado | 4 | 13,3 |
| Total | 30 | 100,0 |

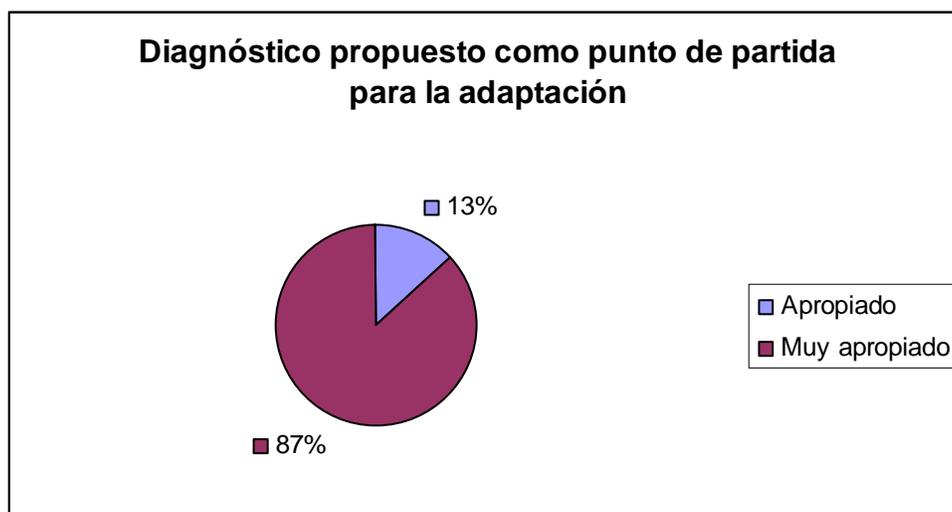


Fig. 2.22 Sobre resultados Tabla 2-6.

Con respecto a la distribución de las tareas para cada agente, fue considerada correcta por el 73.3 % mientras que un 26.6 % determinó que fue medianamente correcta.

Tabla 2-7: Respuesta de los expertos a la pregunta 4 del anexo 3.

| | Frecuencia | Por ciento |
|-----------------------|------------|------------|
| Correcta | 22 | 73,3 |
| Medianamente correcta | 8 | 26,6 |
| Total | 30 | 100,0 |

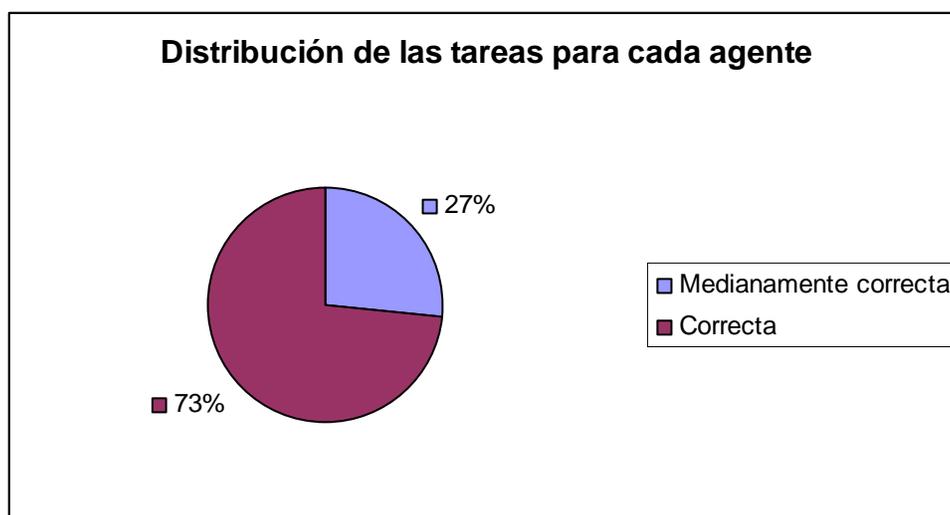


Fig. 2.23 Sobre resultados Tabla 2-7.

La utilidad de proporcionar adaptabilidad a otros sistemas fue considerada por el 83.3 % como “Sí” mientras que el 16.7 % opina que “no sabe”.

Tabla 2-8: Respuesta de los expertos a la pregunta 5 del anexo 3.

| | Frecuencia | Por ciento |
|-------|------------|------------|
| Sí | 25 | 83.3 |
| No sé | 5 | 16.7 |
| Total | 30 | 100,0 |

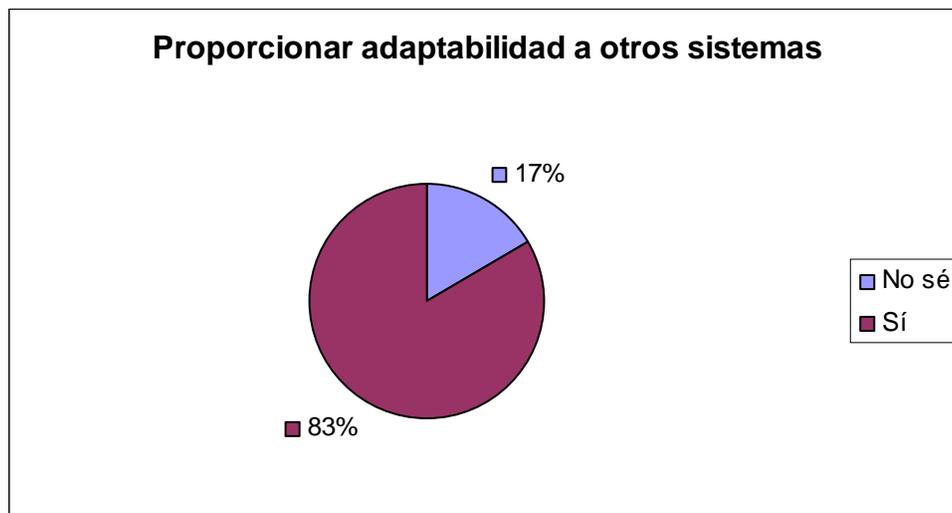


Fig. 2.24 Sobre resultados Tabla 2-8.

2.5 Conclusiones parciales

De esta forma se logró diseñar e implementar un sistema multiagentes que proporciona adaptabilidad a APA-Prolog y es materializado en una aplicación que permite a estudiantes y profesores acceder de una forma rápida y sencilla a la información necesaria para su preparación. Se construyó también una herramienta de administración que permite de un modo fácil y seguro suministrar y actualizar el contenido que se muestra en el sitio principal, así como una aplicación para trabajar con el diagnóstico de los educandos en los diferentes conceptos. La validación realizada por los expertos permitió comprobar que el conjunto de agentes inteligentes proporciona la adaptabilidad necesaria a APA-Prolog para contribuir a personalizar la atención a los estudiantes.

CONCLUSIONES

- ✓ Los agentes inteligentes constituyen una alternativa para lograr ambientes de enseñanza-aprendizaje con posibilidades de adaptarse a las características de los estudiantes. Zope, MY SQL y PHP son herramientas que facilitan y garantizan una implementación adecuada de sistemas para la enseñanza usando técnicas de IA como son los AI.
- ✓ Se implementó un sistema multiagentes que proporciona adaptabilidad a APA-Prolog. Se construyó además una herramienta de administración que permite de un modo fácil y seguro suministrar y actualizar el contenido que se muestra en el sitio principal, así como una aplicación para trabajar con el diagnóstico de los educandos en los diferentes conceptos.
- ✓ La validación realizada por los expertos permitió comprobar que el conjunto de agentes inteligentes proporciona la adaptabilidad necesaria a APA-Prolog para contribuir a personalizar la atención a los estudiantes.

RECOMENDACIONES

- ✓ Establecer una interfaz de humano para los agentes que lo requieran, logrando así que los mismos interactúen con los usuarios.
- ✓ Crear una herramienta que permita generar automáticamente de una forma más fácil los diferentes agentes.

REFERENCIAS BIBLIOGRÁFICAS

[ALM05] Almeida S., Febles J., Bolaños O. Evolución de la enseñanza asistida por computadoras. 2005. Disponible en:

http://www.bvs.sld.cu/revistas/ems/vol11_1_97/ems05197.htm

[ALM07] Almeida, S. Metodología para la gestión del conocimiento en Ciencias Básicas Biomédicas con el empleo de las Tecnologías de la Información y las Comunicaciones. Tesis de Doctorado. Universidad de Matanzas "Camilo Cienfuegos", Matanzas, 2007.

[BAR99] Baray, Cristobal y Wagner, Kyle (1999) **¿Where Do Intelligent Agents Come From?** En: www.acm.org/crossroads/xrds5-4/dumbagents.html . Consultado: 20 septiembre 2005.

[BEL02] Bell Rodríguez, Rafael y López Machín, Ramón (2002). **Convocados por la diversidad. Editorial Pueblo y Educación.** ISBN 959-13-0921-X.

[BEN94] Benyon D. "Accommodating Individual Differences through an Adaptative User Interface" Presented by Alison Nichols, October 21, 1994.

[BER04] Berlanga Flores A. J., García Peñalvo F. J. Sistemas Hipermedia Adaptativos en el ámbito de la Educación. Departamento de Informática y Automática. Universidad de Salamanca. Abril, 2004. Disponible en: -----

[BET06] Betancourt Torres, Juana y otros (2006). Fundamentos psicológicos y pedagógicos de la Educación Especial. Editorial Pueblo y Educación. ISBN 959-13-1439-6.

[BOT02] Botia Blaya, Juan A. (2002). Introducción a los agentes software-MAS. DIIC. Universidad de Murcia. En:

<http://ants.dif.um.es/staff/juanbot/ml/files/20012002/agentes.pdf>.

Consultado:15 noviembre de 2006.

[BRU96] Brusilovsky, P. (1996): Methods and techniques of adaptive hipermedia", en User Modeling and User Adapted Interaction. Vol. 6, no. 2-3, p. 87-129.

- [BRU98] Brusilovsky, P., Eklund, J., Schwarz, E. (1998) "Web-based education for all: A tool for developing adaptive courseware" en Proceedings of Seventh International World Wide Web Conference. Computer Networks and ISDN System, vol. 30 no. 1-7, pp.291-30. Disponible en: www7.scu.edu.au/programme/fullpapers/1893/com1893.htm .Revisado en Octubre de 2002.
- [BUL02] Bull S., Nghiem T. (2002) Helping Learners to Understand themselves with a Learner Model Open to Students, Peers and Instructors, Brna P. and Dimitrova V. (Eds.), Proceedings of Workshop on Individual and Group Modelling Methods that Help Learners Understand Themselves, ITS2002, pp. 5-13.
- [CHO02] CHOUA, C., CHANB T. y LINC C. (2002). Redefining the learning companion: the past, present, future of educational agents, Computer & Education, 40(3), p. 255-269.
- [COL88] Collins A. and Brown J. S. (1988): The computer as a Tool for Learning Through Reflection, in Mandl H. & Lesgold A. (eds), Learning Issue for Intelligent Tutoring Systems, Springer-Verlag, New York, pp 1-18.
- [COS05] Costa, G., Salgueiro, F.A., Cataldi, Z., García, R., Lage, F.J. (2005) Sistemas Inteligentes aplicados al modelo del estudiante.
- [DEB99] De Bra, P., Brusilovsky P., Houben, G. (1999) "Adapative Hipermedia: From Systems to Framework" en ACM Computing Surveys, Symposium Issue on Hypertext and Hypermedia, New York: ACM press, vol. 31, no. 4 pp. 12.
- [EKL97] Eklund, J., Brusilovsky, P., Schwarz, E. (1997): "Adaptive Textbooks on the WWW", en: H. Ashman, P., Thistewaite, R., Debreceny, A. Ellis (eds.) Proceedings of AUSWEB97, 3rd Australian Conference on the World Wide Web. Queensland, Australia. July 5-9. Southern Cross University Press, pp.186-192.
- [ETC98] ETCEE (1998). Ki for JavaWorld. En: <http://www.javaworld.com>. Consultado: 9 octubre 2006.
- [FIS00] Fischer, G. (2000) User Modeling in Human-Computer Interaction. Contribution to the 10th Anniversary Issue of the Journal "User Modeling and User-Adapted Interaction (UMUAI)".

- [GAL94] Galvis, P. A. (1994) Ingeniería de Software Educativo. Ediciones Uniandes, Santafé de Bogotá.
- [GAR00] García, Z. Hipermedia para la enseñanza de las estructuras básicas de control de la programación estructurada. in V Congreso Iberoamericano de Informática Educativa. Chile. 2000
- [GAR03] García Serrano, Ana M. y Hernández, Josefa Z. (2003). Agentes Inteligentes y Sistemas Multiagente. Doctorado en Ciencias de la Computación e Inteligencia Artificial. Departamento de Inteligencia Artificial. Curso 2002-2003
- [GAU02] Gaudioso E. (2002) “Contribuciones al modelado del usuario en entornos adaptativos de aprendizaje y colaboración a través de Internet mediante técnicas de aprendizaje automático”. Tesis doctoral. Universidad de Educación a Distancia. Madrid. España. Disponible en: www.ia.uned.es/personal/elena/egvtesis.pdf
- [GOL95] Goleman D. (1995): Emocional Intelligence, New York: Bantam Books.
- [GON03] Gonzáles C. Sistemas Inteligentes en la Educación: Una revisión de las líneas de investigación y aplicaciones actuales. 2003 Disponible en: http://www.uv.es/RELIEVE/v10n1/RELIEVEv10n1_1.htm
- [GON07] González M., Simulador de Backtracking para el trabajo con listas en Prolog. Tesis en opción al grado científico de Master en Nuevas Tecnologías para la Educación. Universidad de S.S “José Martí”, 2007.
- [GRE97] Green, Shaw and others. **Software Agents: A review.** En: www.cs.tcd.ie/research_groups/aig/iag/iag.html. 1997.
- [GRI08] Grimón, F. Modelo para la gestión de dominios de contenido en sistemas hipermedia adaptativos aplicados a entornos de Educación Superior., Universidad Politécnica de Cataluña, Barcelona. España, 2008.
- [HAY02] Hayashi, Hisashi and others. **Mobile Agents and Logic Programming.** Computer and Network Systems Laboratory Corporate Research and Development

Center TOSHIBA CORPORATION, Japan. En: LNCS25352002Mobile Agents : 6th International Conference, MA 2002, Barcelona, Spain, October 22-25 Proceedings

[HEN03] Henze, N. (2003): "From Web-Based Educational System to Education on the Web: on the Road to the Adaptive Web", en Vasile Palade, Robert J. (Eds.): Knowledge-Based Intelligent Information And Engineering System, 7th International Conference, KES 2003, Oxford, Reino Unido, LNCS, vol. 2774 Springer. Pp. 297-303.

[HER99] Hernan Tolosa, and others (1999). **Revisión: tecnología de agentes de software.** Departamento de Ciencias Básicas, Universidad Nacional de Luján, Argentina. En: Rev. Ci. Inf., Brasília, v. 28, n. 3, p. 302-309, set.-dez.

[HER05] Hernansáez Amor, and others (2005). **Asistencia personalizada a la Minería de Datos mediante Agentes Inteligentes.** Departamento de Ingeniería de la Información y las Comunicaciones. Universidad de Murcia.

[HOL94] Holt P., Dubs S., Jones M., Greer J. (1994) "The state of student modelling", en J. E. Greer y G. I. McCalla, (eds), Student modelling: The key to individualized knowledge-based instruction. NATO ASI Series, vol. 125, Springer-Verlag. pp 3-38.

[JUL00] Julian, V. and Botti, V. **Agentes Inteligentes: el siguiente paso en la Inteligencia Artificial.** Universidad Politécnica de Valencia. España. En: Rev Novatica, edición digital. 2000.

[KOB01] Kobsa, A., Koenemann, J., Pohl, W. (2001): "Personalized Hypermedia Presentation Techniques for Improving Online Customer Relationships", en The knowledge Engineering Review, vol. 16, no.2 pp. 111-1555.

[LEZ98] Lezcano M., Ambientes de aprendizaje por descubrimiento para la disciplina Inteligencia Artificial. Santa Clara. 1998.

[LOP07] López, M. Modelo pedagógico del tutor en la enseñanza-aprendizaje del inglés a distancia con el aprovechamiento de las TIC en el postgrado. Tesis de Doctorado. Instituto Superior Pedagógico "Enrique José Varona" Facultad de Lenguas Extranjeras, Ciudad de La Habana. Cuba, 2007.

- [RIO10] Ríos, L. Ambiente Inteligente de Enseñanza-Aprendizaje para Prolog. Memorias del 7mo Congreso Internacional de Educación Superior. La Habana, Cuba, 2010.
- [MAR07] Martínez A., Simulador de Bactracking para el Prolog. Tesis en opción al grado científico de Master en Nuevas Tecnologías para la Educación. Universidad de S.S “José Martí”, 2007.
- [MED07] Medina, D., Acosta, A., Martinez, N., Garcia, Z. (2007a) MacBay: Herramienta para construir Sistemas de Enseñanza-Aprendizaje Inteligentes con un enfoque bayesiano. Presentado en COMPUMAT 2007.
- [MED07a] Medina D. Redes Bayesianas y Mapas Conceptuales en la elaboración de Sistemas de Enseñanza-Aprendizaje Inteligentes. Trabajo para optar por el Título de Máster en Computación Aplicada. UCLV, 2007.
- [MAE94] Maes, Patty. **Modeling Adaptive Autonomous Agents**. MIT Media-Laboratory, Cambridge. 1994.
- [MOU07] Mouriz Y., Simulador para el aprendizaje del proceso de matching. Tesis en opción al grado científico de Master en Nuevas Tecnologías para la Educación. Universidad de S.S “José Martí”, 2007.
- [OKA00] Okamoto T., Cristea A. I., Kayama M. Towards Intelligent Media-Oriented Distance Learning and Education Environments. Proceedings of ICCE2000 Computer-Supported Collaborative Learning, Springer Verlag, pp. 69-97, 2000.
- [PEÑ04] Peña A., Gutiérrez F. Propuesta conceptual para un sistema adaptativo de educación basada en Web. 8vo Congreso Nacional de Ingeniería Electromecánica y de Sistemas. México. 2004.
- [PEÑ04a] Peña A., Sossa J. H. Educación basada en Web, Modelo del Estudiante y Mapas Cognitivos: Estado del Arte. Instituto Politécnico Nacional de México. No. 108 Serie: Verde. ISBN 970-36-0170-7. Año 2004.
- [PEÑ04b] Peña, C. Intelligent agents to improve adaptivity in a web-based learning environment. Tesis de Doctorado. Department of electronics, computer science and automatic control. University of Girona, 2004.

- [PER03] Perez A., Florido R. Posibilidades y limitaciones de Internet como recurso educativo. Publicación en línea, Año 1, No. 2, ISSN: 1695-324X, 2003.
<http://www.ugr.es/~sevimeco/revistaeticanet/index.htm>
- [PER05] Pérez R, López A. Los retos de la enseñanza Virtual. La experiencia AulaNet.
<http://www.aulanet.uniovi.es/portal/publicaciones/documentos/XIVAsepeltOviedo/LosRetos.htm>
- [PIA89] Piaget J. (1989): Les relations entre l'intelligence et l'affectivité dans le développement de l'enfant, Rimé B. ;Scherer, K. (Eds.) Les Emotions. Textes de base en psychologie. Paris : Delachaux et Niestlé, pp. 75-95.
- [PSN01] Primer Seminario Nacional de Elaboración de guiones de Software Educativo para a Escuela Cubana. La Habana. 2001.
- [ROD07] Rodríguez M., Directorio Temático para Prolog. Tesis en opción al grado científico de Master en Nuevas Tecnologías para la Educación. Universidad de S.S "José Martí", 2007.
- [ROJ03] Rojas A., La Comunicación Educativa en las condiciones de la Educación a Distancia. Revista Pedagogía Universitaria. No. 3, Vol. 8, 2003.
- [RUS95] Russell, S. y Norvig, Peter. **Artificial Intelligence: A Modern Approach**. Prentice-Hall. 1995.
- [SEV98] Sevillano, M.L. (1998) Nuevas Tecnologías, Medios de Comunicación y Educación. Formación inicial y permanente del profesorado. España: Editorial CCS.
- [SCH96] Schwarz E., Brusilovsky P., Weber, G. (1996): "World-Wide Intelligent Textbooks", en Proceedings of the World Conference on Educational Telecommunications ED-TELECOM'96. pp. 302-307. Disponible en:
www.contrib.andrew.cmu.edu/~ED-MEDIA-96.html Revisado: 10 de marzo de 2003.
- [SHA05] Shalfield, Rebecca (2005). **Agent Toolkit**. Documentación del software LPA WIN_ PROLOG.
- [SOL05] Soledad C., Sistemas Inteligentes en la Educación: Una Revisión de las Líneas de Investigación y Aplicaciones Actuales. Disponible en:
http://www.uv.es/RELIEVE/v10n1/RELIEVEv10n1_1.htm

[SPE97] Specht, M., Weber, G., Heitmeyer, S., Schöch, V. (1997): “AST: Adaptive WWW-Courseware for Statistics”, en P. Brusilovsky, J. Fink, & J. Key (Eds.) Proceedings of Workshop Adaptive Systems and User Modeling on the World Wide Web 6th Internacional Conference on User Modeling, UM97. Sardinia, Italia, junio. Carnegie Mellon Online. Pp. 91-95. Disponible en:

www.contrib.andrew.cmu.edu/~plb/UM97workshop/Specht.html

Revisado: 10 de marzo de 2003.

[STE92] Steels, Luc. (1992) **A case study in the behavior-oriented design of autonomous agents**. Artificial Intelligence Laboratory. Vrije Universiteit Brussel, Belgium.

[VAL87]. Valdés Pardo V. G.: Investigación y elaboración de Sistemas para la enseñanza asistida por computadora. Tesis de Dr. En Ciencias Técnicas, Santa Clara, 1987, pp. 19-26.

[VYG78] Vygotsky L. S. (1978): Mind in Society: the Development of Higher Psychological Process, London Harvard University Press.

[VIZ98] Vizcarro C., León J. Nuevas Tecnologías para el aprendizaje. Madrid. 1998.

[WEB01] Weber G., Brusilovsky, P. (2001): “ELM-ART: An Adaptive Versatile System for Web-based Instruction”. International Journal of Artificial Intelligence in Education (12). Pp. 351-384.

[LAR99] Larman, C. Introducción al análisis y diseño orientado a objetos, Prentice Hall. 1999.

[SOF09] Software Engineering Institute, Calidad, disponible en:

<http://www.sei.com/html/esp/calidad.html>, visitado: 14/01/09.

[CUB08] Cubells, V. Zope, una alternativa para el desarrollo de aplicaciones Web. Cuba, disponible en:

<http://www.cujae.edu.cu/revistas/telematica/Articulos/103.htm> visitado: (12/11/2008)

[ART09]_Arteaga, L. La licencia Pública como alternativa al Software Propietario. Disponible en:

http://www.aui.es/biblio/bolet/bole024/art_gnu.htm Revisado: (13/01/2009)

[POR09] ¿Por que usar Software libre?, disponible en:

http://www.micorp.com.ve/descargas/files/Software_LibreV2.pdf

(15/01/2009)

Revisado:

- [PER08] Pérez, J. Las Tecnologías Web para la Gestión del Conocimiento. Cuba, disponible en:
http://www.sociedadelainformacion.com/9/las_tecnologias_web.htm
Visitado:(14/11/2008)
- [ZOP08] Zope: mucho más que un servidor Web, disponible en:
<http://jungla.dit.upm.es/~joaquin/las/trabajos/2001/ZOPE.pdf.gz> Visitado: (06/11/2008)
- [ZOP09] ZopeBook Versión 2.6 traducción al español, disponible en:
<http://glud.udistrital.edu.co/glud/proyectos/zope/x196.html> Visitado: (09/01/2009)
- [MYS08] MySQL AB (2005). Database Server Feature Comparisons, disponible en:
<http://dev.mysql.com/tech-resources/features.html> Vistado: (10/12/2008)
- [POR08] ¿Por qué usar CSS?, disponible en:
http://www.adobe.com/es/devnet/dreamweaver/articles/why_css.html Visitado:
(10/12/2008)
- [MED07] Medina, D. (2007). Redes Bayesianas y Mapas Conceptuales en la elaboración de Sistemas de Enseñanza-Aprendizaje Inteligentes. Tesis de Maestría. Universidad Central "Marta Abreu" de Las Villas.
- [FIS00] Fischer, G. (2000). User Modeling in Human-Computer Interaction. Contribution to the 10th Anniversary Issue of the Journal "User Modeling and User-Adapted Interaction (UMUAI)".
- [DEF09] Definición de Teoría del Aprendizaje de Vigotsky. Disponible en:
<http://www.psicopedagogia.com/definicion/teoria%20del%20aprendizaje%20de%20vigotsky> Visitado: 07/2009
- [WIK09] Wikipedia Enciclopedia Libre, *Pedagogía*. Disponible en:
<http://es.wikipedia.org/wiki/Pedagog%C3%ADa>, Visitado 10/2009.
- [VIG78] Vigotsky, L.S., *Mind in Society: the Development of Higer Psychological Process*. London Harvard University Pres, 1978.
- [TEO09] Vigotsky, L.S., *La teoría socio histórica de Lev Semenovich Vigotsky*. Disponible en <http://vigotsky.idoneos.com/>, Visitado:07/2009.
- [WEI99] Weiss, G. Multiagent Systems. A Modern Approach to Distributed Modern Approach to Artificial Intelligence. Prentice-Hall Editor, 1999.

GLOSARIO

Apache – Servidor de páginas Web de código abierto para diferentes plataformas (UNIX, Windows, etc.).

CSS (Cascading Style Sheets) – Las hojas de estilo en cascada contienen un conjunto de etiquetas que definen el formato que se aplicará al contenido de las páginas de una Web. Se llaman “en cascada” porque una hoja puede heredar los formatos definidos en otra hoja de forma que no hace falta que vuelva a definirlos.

HTML (HyperText Markup Language) – Lenguaje basado en marcas que indican las características del texto, utilizado para definir documentos de hipertexto en páginas Web.

HTTP (HyperText Transfer Protocol) – Protocolo cliente-servidor utilizado para el intercambio de páginas Web (HTML).

MySQL – Sistema multiplataforma Open Source de gestión de bases de datos relacionales, muy popular en aplicaciones Web.

Open source – Código abierto o código libre. Software que distribuye de forma libre su código fuente, de forma que los desarrolladores pueden hacer sus variaciones, mejoras o reutilizarlo en otras aplicaciones. También conocido como free software.

Perl (Practical Extraction and Report Language) – Lenguaje para manipular textos, ficheros y procesos. Con estructura de script. Desarrollado por Larry Wall, es multiplataforma ya que funciona en Unix.

Python – Lenguaje de programación interpretado e interactivo, capaz de ejecutarse en una gran cantidad de plataformas. Fue creado por Guido van Rossum en 1990.

Web – Telaraña, malla.- La palabra "Web" se utiliza para denominar uno de los servicios más importantes de la red Internet. Son páginas que utilizan un lenguaje especial llamado HTML, que permite presentar en pantalla texto y gráficos en el formato deseado. Estas páginas contienen referencias o enlaces que permiten acceder a otras páginas. Existen millones de páginas Web con gran cantidad de información sobre todo tipo de temas.

WebDAV (Web-based Distributed Authoring and Versioning) – Es una extensión del protocolo HTTP que permite a los usuarios editar y administrar ficheros de forma colaborativa en servidores Web remotos.

Frames – Elementos HTML que permiten al navegador separar su ventana en unidades independientes para recibir código HTML distinto en cada una.

UML (Unified Modelling Language) – Es el lenguaje de modelado de sistemas de software más conocido en la actualidad.

ZOPE – Servidor de aplicaciones Web escrito en el lenguaje de programación Python. Puede ser manejado casi totalmente usando una interfase de usuario basada en páginas Web.

ZMI – Interfase de administración de ZOPE.

ACL (Access Control List) – Lista de control de acceso que limita el uso específico de recursos a usuarios autorizados.

LDAP (Lightweight Directory Access Protocol) – Servicio de directorio ordenado y distribuido para buscar diversa información en un entorno de red. LDAP puede

considerarse una base de datos (aunque su sistema de almacenamiento puede ser otro diferente) al que pueden realizarse consultas.

RADIUS (Remote Access Dial-In User Server) – Protocolo de autenticación y autorización para aplicaciones de acceso a la red o movilidad IP.

SMB (Server Message Block) – Protocolo de red que permite compartir archivos e impresoras, utilizado en redes Microsoft Windows.

PostgreSQL – Es un servidor de base de datos relacional libre, liberado bajo la licencia BSD. Es una alternativa a otros sistemas de bases de datos de código abierto (como MySQL, Firebird y MaxDB), así como sistemas propietarios como Oracle o DB2.

PHP (PHP: Hypertext Preprocessor) – Originado inicialmente del nombre PHP Tools, o Personal Home Page Tools. Es un lenguaje de programación interpretado. Aunque fue concebido en el tercer trimestre de 1994 por Rasmus Lerdorf no fue hasta el día 8 de Junio de 1995 que fue lanzada la versión 1.0. Se utiliza entre otras cosas para la programación de páginas Web activas y se destaca por su capacidad de mezclarse con el código HTML.

GNU (Gnu is Not Unix) – Conjunto de programas desarrollados por miembros de la Free Software Foundation (Fundación por el Software Libre); son de uso gratuito.

Firefox – Navegador Web del proyecto Mozilla. El objetivo de Firefox es desarrollar un nuevo navegador para el conjunto Mozilla suite, aunque también el desarrollo de un navegador independiente de la suite, que fuera más pequeño, rápido y que aportara características novedosas a la navegación. Del mismo modo que Mozilla Suite, Firefox es multiplataforma, utiliza el lenguaje de interfaz XUL y es software libre.

FTP (File Transfer Protocol) – Protocolo de Transferencia de Archivos, permite trasladar archivos de un computador a otro. Es un protocolo estandarizado de Internet.

Plug-ins – Aplicaciones software que se pueden instalar en el mismo software de edición no lineal para proporcionar al sistema funciones y características adicionales.

Netscape – Navegador que facilita el acceso a la red. Hoy en día es uno de los más utilizados que consiguió desbancar al Mosaic, el primero que empezó a utilizarse para visualizar las páginas de hipertexto.

CPU (Central Processing Unit) – Unidad donde se ejecutan las instrucciones de los programas y se controla el funcionamiento de los distintos componentes del ordenador. Suele estar integrada en un chip denominado microprocesador.

JavaScript – Lenguaje interpretado orientado a las páginas Web, con una sintaxis semejante a la del lenguaje Java.

W3C (World Wide Web Consortium) – Organización que produce estándares para la World Wide Web (o Telaraña Mundial). Está dirigida por Tim Berners-Lee, el creador original de URL (Uniform Resource Locator, Localizador Uniforme de Recursos), HTTP (HyperText Transfert Protocol, Protocolo de Transferencia de HiperTexto) y HTML (Lenguaje de Marcado de HiperTexto) sobre el que se basa la Web, y por extensión su inventor.

Combobox – Objeto compuesto por una lista, en la que el usuario puede elegir una de las opciones que ofrece, y una ventana de texto en la que el usuario puede introducir una expresión que no está en la lista..

ANEXOS

Anexo #1. Encuesta para la selección de los expertos que evalúan el Ambiente de Enseñanza-Aprendizaje para la Programación Lógica (APA-Prolog)

Método Delphi. Procedimiento de autovaloración de los expertos.

Agradeciéndole de antemano su colaboración en esta encuesta le solicitamos autoevalúe sus competencias como experto que dictaminará sobre el Ambiente de enseñanza-aprendizaje para la Programación Lógica (APA-Prolog).

Utilizaremos el procedimiento de autovaloración del método Delphi:

- a) Evalúe su competencia sobre el problema en una escala de 0 a 10. En esta escala el 0 representa que el experto no tiene competencia alguna sobre el problema y el 10 representa que posee una información completa sobre el tema.

| | | | | | | | | | | |
|--------------------------|--------------------------|--------------------------|--------------------------|--------------------------|--------------------------|--------------------------|--------------------------|--------------------------|--------------------------|--------------------------|
| <input type="checkbox"/> |
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |

- b) Marque, por favor, con una cruz cuál de las fuentes considera que ha influido en su conocimiento de acuerdo con el grado A, M o B.

| FUENTES DE ARGUMENTACION | Grado de influencia de cada una de las fuentes en sus criterios. | | |
|--|--|--------------|-------------|
| | A (alto) | M (medio) | B (bajo) |
| Conocimiento sobre diseño de agentes y sistemas multiagentes. | | | |
| Su experiencia como profesor o profesional en el área de Programación de Computadoras | | | |
| Conocimiento sobre trabajos de autores nacionales y extranjeros relacionados con el tema | | | |
| Conocimientos teóricos sobre la Programación Lógica. | | | |
| Conocimiento sobre metodología de enseñanza de la Programación | | | |
| Conocimiento sobre diseño de software educativo | | | |

Los autores le quedan agradecidos por su colaboración y le garantizamos el anonimato.

Anexo #2. Determinación del nivel de competencia de los expertos y distribución por categorías docentes y científicas

| EXP. # | KC | KA | K | | F1 | F2 | F3 | F4 | F5 | F6 |
|--------|-----|-------|--------|-------|------|------|-------|-------|------|------|
| 1 | 0,8 | 0,95 | 0,875 | ALTA | 0,2 | 0,4 | 0,05 | 0,04 | 0,16 | 0,1 |
| 2 | 0,8 | 0,76 | 0,78 | MEDIA | 0,16 | 0,32 | 0,04 | 0,04 | 0,1 | 0,1 |
| 3 | 0,8 | 0,89 | 0,845 | ALTA | 0,2 | 0,4 | 0,05 | 0,04 | 0,1 | 0,1 |
| 4 | 0,7 | 0,88 | 0,79 | MEDIA | 0,2 | 0,4 | 0,04 | 0,04 | 0,1 | 0,1 |
| 5 | 0,8 | 0,82 | 0,81 | ALTA | 0,16 | 0,32 | 0,04 | 0,04 | 0,16 | 0,1 |
| 6 | 0,8 | 0,89 | 0,845 | ALTA | 0,16 | 0,4 | 0,05 | 0,04 | 0,16 | 0,08 |
| 7 | 0,8 | 0,92 | 0,86 | ALTA | 0,2 | 0,4 | 0,04 | 0,04 | 0,16 | 0,08 |
| 8 | 0,9 | 0,915 | 0,9075 | ALTA | 0,2 | 0,4 | 0,05 | 0,025 | 0,16 | 0,08 |
| 9 | 1 | 0,885 | 0,9425 | ALTA | 0,16 | 0,4 | 0,04 | 0,025 | 0,16 | 0,1 |
| 10 | 1 | 0,89 | 0,945 | ALTA | 0,16 | 0,4 | 0,025 | 0,025 | 0,2 | 0,08 |
| 11 | 1 | 0,99 | 0,995 | ALTA | 0,2 | 0,4 | 0,05 | 0,04 | 0,2 | 0,1 |
| 12 | 0,9 | 0,88 | 0,89 | ALTA | 0,16 | 0,4 | 0,04 | 0,04 | 0,16 | 0,08 |
| 13 | 0,6 | 0,635 | 0,6175 | MEDIA | 0,1 | 0,32 | 0,04 | 0,025 | 0,1 | 0,05 |
| 14 | 1 | 0,805 | 0,9025 | ALTA | 0,16 | 0,32 | 0,025 | 0,04 | 0,16 | 0,1 |
| 15 | 1 | 0,98 | 0,99 | ALTA | 0,2 | 0,4 | 0,04 | 0,04 | 0,2 | 0,1 |
| 16 | 0,7 | 0,98 | 0,84 | ALTA | 0,2 | 0,4 | 0,04 | 0,04 | 0,2 | 0,1 |
| 17 | 1 | 0,92 | 0,96 | ALTA | 0,16 | 0,4 | 0,04 | 0,04 | 0,2 | 0,08 |
| 18 | 1 | 0,74 | 0,87 | ALTA | 0,16 | 0,32 | 0,025 | 0,025 | 0,16 | 0,05 |
| 19 | 1 | 0,98 | 0,99 | ALTA | 0,2 | 0,4 | 0,05 | 0,05 | 0,2 | 0,08 |
| 20 | 1 | 0,98 | 0,99 | ALTA | 0,2 | 0,4 | 0,05 | 0,05 | 0,2 | 0,08 |
| 21 | 0,8 | 0,86 | 0,83 | ALTA | 0,2 | 0,4 | 0,04 | 0,04 | 0,1 | 0,08 |
| 22 | 0,6 | 0,785 | 0,6925 | MEDIA | 0,2 | 0,32 | 0,04 | 0,025 | 0,1 | 0,1 |
| 23 | 0,8 | 1 | 0,9 | ALTA | 0,2 | 0,4 | 0,05 | 0,05 | 0,2 | 0,1 |
| 24 | 0,8 | 0,9 | 0,85 | ALTA | 0,16 | 0,4 | 0,04 | 0,04 | 0,16 | 0,1 |
| 25 | 0,8 | 0,885 | 0,8425 | ALTA | 0,16 | 0,4 | 0,04 | 0,025 | 0,16 | 0,1 |
| 26 | 0,8 | 0,79 | 0,795 | MEDIA | 0,16 | 0,32 | 0,025 | 0,025 | 0,16 | 0,1 |
| 27 | 1 | 0,99 | 0,995 | ALTA | 0,2 | 0,4 | 0,05 | 0,04 | 0,2 | 0,1 |
| 28 | 0,6 | 0,53 | 0,565 | MEDIA | 0,1 | 0,2 | 0,025 | 0,025 | 0,1 | 0,08 |
| 29 | 0,9 | 0,965 | 0,9325 | ALTA | 0,2 | 0,4 | 0,04 | 0,025 | 0,2 | 0,1 |
| 30 | 0,9 | 0,97 | 0,935 | ALTA | 0,2 | 0,4 | 0,04 | 0,05 | 0,2 | 0,08 |

F1-F6: Fuentes de Argumentación.

KC: es el coeficiente de conocimiento o información que tiene el experto acerca del problema calculado sobre la base de la valoración del propio experto

KA: es el coeficiente de argumentación o fundamentación de los criterios del experto determinado como resultado de la suma de los puntos alcanzados a partir de la tabla patrón (Anexo 8):

K: Coeficiente medio.

| Categoría Docente | Cantidad | % |
|-----------------------------|----------|------|
| Titulares | 16 | 53.3 |
| Auxiliares | 14 | 46.7 |
| Categoría Científica | | |
| Doctores | 22 | 73.3 |
| Master | 8 | 26.7 |

Anexo #3. Encuesta a los expertos.

En el Centro Universitario de Sancti Spiritus “José Martí” se desarrolló un ambiente de enseñanza- aprendizaje para la Programación Lógica (APA-Prolog), al cual se le han incorporado un grupo de agentes inteligentes que proporcionan la adaptabilidad necesaria para personalizar la atención a los estudiantes. En este momento pretendemos recoger opiniones acerca de la calidad de este recurso. Hemos pensado en usted por el dominio que tiene sobre el tema, le sugerimos que al revisar el software fije su atención en el contenido, la armonía, el tratamiento diferenciado a los estudiantes y en la conveniencia de extender la experiencia a otras asignaturas. Le agradecemos de antemano su colaboración, le garantizamos el anonimato y le pedimos que sea sincero, considerando la importancia de sus criterios para perfeccionar este trabajo.

1. ¿Considera que a partir de este ambiente es posible elevar los niveles de asimilación de la Programación Lógica?
 muy adecuado
 adecuado
 ni adecuado ni inadecuado
 inadecuado
 muy inadecuado
2. Su valoración acerca de las posibilidades que tiene el ambiente de tomar en cuenta los conocimientos previos del estudiante es:
 muy buena
 buena
 ni buena ni mala
 mala
 muy mala
3. ¿Cómo considera el diagnóstico propuesto como punto de partida para la adaptación?
 muy apropiado
 apropiado
 ni apropiado ni inapropiado
 inapropiado
 muy inapropiado
4. ¿Cómo valora usted la distribución de las tareas para cada agente?
 correcta
 medianamente correcta
 incorrecta
5. ¿Considera usted útil proporcionar adaptabilidad a otros sistemas para la enseñanza?
 sí
 no
 no sé