



**Centro Universitario de Sancti Spiritus**

**“José Martí Pérez”**

**Tesis en Opción al Grado Científico de Máster en  
Nuevas Tecnologías para la Educación**

**Título:** Simulador para el aprendizaje del tema Listas en la  
asignatura de Programación Lógica.

**Autor:** Ing. Maité González Cruz.

**Tutor:** MSc. Lydia Rosa Ríos

Junio del 2007

“Año 49 de la Revolución”

**Resumen**

En esta investigación se presentan los resultados de un estudio para mejorar el proceso de aprendizaje del tema Listas en la asignatura Programación Lógica (PROLOG) para estudiantes de la carrera de Lic. en Ciencias de la Computación, problemática que ha sido preocupación de docentes y estudiantes de dicha carrera de la Universidad Central “Marta Abreu de las Villas”. Sobre la base de los estudios diagnóstico se propone implementar un simulador para mejorar el aprendizaje de dicho tema. En el proceso investigativo quedaron explicados un conjunto de métodos del nivel empírico tales como la observación, la revisión de documentos, la encuesta y la entrevista. Del nivel teórico el análisis-síntesis, la inducción-deducción, el análisis histórico-lógico y la modelación; y del nivel estadístico se utilizó el cálculo porcentual. Estos métodos fueron aplicados a una población conformada por los 36 estudiantes de 4to año de la carrera de Lic en Ciencias de la Computación de la Universidad Central de las Villas, coincidiendo con la muestra. La propuesta fue sometida a la validación por el criterio de expertos consultándose a 7 especialistas los que llegaron a la conclusión de la pertinencia y factibilidad de la propuesta para perfeccionar el proceso de aprendizaje de los alumnos.

<b>Introducción</b>	<b>1</b>
<b>Capítulo 1: Nuevo reto de las TIC y su impacto en la Programación Lógica.</b>	<b>9</b>
<b>1.1 Las Tecnologías de la Información y la Comunicación</b>	<b>9</b>
<b>1.2 Impacto de las nuevas tecnologías en los sistemas educativos.</b>	<b>10</b>
<b>1.3 La formación del profesional de la rama de la Informática.</b>	<b>12</b>
<b>1.4 Algunos criterios sobre la Programación Lógica</b>	<b>14</b>
<b>1.5 Listas en la Programación Lógica</b>	<b>15</b>
<b>1.6 Reflexiones sobre los Software Educativos.</b>	<b>16</b>
1.6.1 Otras consideraciones acerca de los simuladores.	19
<b>1.7 Los Mapas Conceptuales para el aprendizaje de Prolog.</b>	<b>21</b>
<b>1.8 Técnicas analizadas. Análisis crítico</b>	<b>23</b>
1.8.1 Software Educativo Teaching.	23
1.8.2 Teaching 2.0	25
1.8.3 Características de W-Prolog	27
1.8.4 El lenguaje Java.	29
1.8.5 Easy Java Simulations	31
<b>1.9 Algunas consideraciones acerca del NetBeans</b>	<b>33</b>
<b>Capítulo 2: Propuesta de diseño del Simulador para el aprendizaje del tema Listas en Prolog.</b>	<b>34</b>
<b>2.1 Justificación de la población y muestra.</b>	<b>34</b>
<b>2.2 Métodos y técnicas de la investigación educacional.</b>	<b>35</b>
<b>2.3 Resultado de la aplicación de los instrumentos seleccionados</b>	<b>37</b>
<b>2.4 Fundamento de las tendencias y metodologías en las que se sustenta la propuesta.</b>	<b>43</b>
2.4.1 Tendencias Pedagógicas.	43
2.4.2 Fundamentación de la metodología utilizada.	46
2.4.3 Rational Rose	47
<b>2.5 Descripción del método de trabajo.</b>	<b>47</b>
<b>2.6 Análisis del diseño del sistema.</b>	<b>48</b>
2.6.1 Diagrama de Casos de Uso del Negocio.	48
2.6.2 Diagrama de Clases del Modelo de Objetos.	49
2.6.3 Diagrama de Actividad.	50
2.6.4 Especificación de los requerimientos iniciales del software.	51
2.6.5 Requerimientos no funcionales.	51
2.6.6 Diagrama de casos de uso del sistema.	52
2.6.7 Descripción de los casos de uso del sistema.	53
2.6.8 Diagrama de navegación.	53
2.6.9 Diagrama de interacción.	55
<b>2.7 Valoración de la factibilidad del Simulador para la enseñanza del tema Listas en la asignatura de Programación Lógica por el criterio de expertos.</b>	<b>56</b>
<b>Conclusiones</b>	<b>60</b>
<b>Recomendaciones</b>	<b>61</b>
<b>Referencias Bibliográficas</b>	<b>62</b>
<b>Bibliografía</b>	<b>65</b>
<b>Anexos</b>	<b>69</b>

## Introducción

Desde hace unos años se afirma frecuentemente que vivimos en una "sociedad del conocimiento" o "sociedad de la información" y esto es debido a que gran parte del progreso socioeconómico está asentado en una continua producción de innovaciones tecno-científicas que hacen que la ciencia ocupe definitivamente un papel preponderante en nuestras vidas. Tal es así que a finales del siglo pasado, la casi totalidad de los campos profesionales han visto incrementado su potencial de desarrollo con la incorporación de las nuevas tecnologías y ello ha motivado un cambio sustancial en el modo de ejercer las funciones específicas en cada uno de estos campos.

El investigador Terceiro describe la importancia de la tecnología en el proceso evolutivo de la sociedad al afirmar: "...La humanidad ha venido midiendo su progreso históricamente, en términos de tecnología, con el resultado de que cada era ha sobrepasado más rápidamente que las anteriores. La edad de piedra duró millones de años, pero la siguiente, la de metal, sólo cinco mil años. La revolución industrial, doscientos años. La era eléctrica (de comienzos del siglo XX a la II Guerra Mundial), cuarenta años. La era electrónica duró veinticinco años y la era de la información ya tiene veinte, evolucionando rápidamente desde lo que podríamos llamar Infolítico Inferior al Infolítico Superior o Información hipermedia..." [TER96]

Evidentemente nuevas etapas han aparecido desde el momento en que Terceiro realiza este análisis, pues ya pasamos el momento de los hipermedia y hablamos de "Información en Internet". No obstante sigue estando claro que es esta la sociedad con mayor velocidad de evolución de todas aquellas que hayamos conocido.

Este desarrollo acelerado tecnológico ha dado surgimiento a las Nuevas Tecnologías de la Información y las Comunicaciones (NTIC o bien puede referirse como TIC), las cuales están inundando el mundo referencial del ser humano, a la vez que le están ayudando a conquistar conocimientos y acciones que ayer mismo

parecían inaccesibles pero, de la misma manera, le están condicionando y obligando a adaptaciones y replanteamientos en todos los órdenes de su existencia. [MII05]

El uso de la tecnología de información, la PC e Internet en los procesos de enseñanza y aprendizaje, no puede interpretarse como un medio tecnológico más, sino como un agente de profundos cambios en todo el sistema, que requiere de una buena proyección, planificación y voluntad política. Hoy en día el aprendizaje ha ido más allá de los muros del salón de clases de la universidad convencional, es una realidad que los alumnos y profesores participen en conversaciones intelectuales obviando las barreras geográficas.

El profesor Giraldo Valdés define el software educativo o programas informáticos orientados a fines educacionales como: "...aquél que se destina a apoyar o facilitar diferentes procesos presentes en los sistemas educacionales, entre los cuales cabe mencionar el proceso de enseñanza y aprendizaje, el de vinculación con la práctica laboral, el de investigación estudiantil, el de gestión académica, el de extensión a la comunidad, etc., permitiendo incorporar los sistemas computacionales como medios auxiliares en subsistemas didácticos que abarcan objetivos, contenidos, medios, métodos y evaluación, sobre una o varias disciplinas, en las modalidades presencial, semipresencial o a distancia.". [VAL00]

El desarrollo de la multimedia, el hipertexto y la hipermedia ha permitido la elaboración y explotación de softwares con las facilidades que la combinación de textos, sonidos, imágenes y animaciones pueden contribuir al procesamiento de la información en diferentes campos. [RIO05] Cada día estas técnicas se convierten en un instrumento eficaz de las comunicaciones y el acceso a la información y brindan una enorme ayuda en los procesos de enseñanza-aprendizaje, dígase la educación especial, la educación a distancia y la educación basada en métodos tradicionales.

La disciplina Inteligencia Artificial tiene entre sus asignaturas la Programación Descriptiva en la cual se aborda la Programación Lógica (Prolog).

Actualmente, la Programación Lógica ha despertado un creciente interés que va mucho más allá del campo de la Inteligencia Artificial (IA) y sus aplicaciones. Los japoneses, con sus proyectos de máquinas de la quinta generación, dieron un gran impulso a este paradigma de programación. Sin embargo, antes que ellos existían ya en Estados Unidos y en Europa grupos de investigación en este campo en países como Inglaterra, Holanda, Suecia y, desde luego, Francia. [LOY06]

Prolog es un lenguaje de programación hecho para representar y utilizar el conocimiento que se tiene sobre un determinado dominio. Muchos de los procesos que lo conforman no son visibles pues los realiza la computadora internamente, lo que hace que no resulten del todo entendible al alumno, dificultando el exitoso desarrollo de la asignatura [RIO05].

En la Universidad Central de las Villas se estudia la carrera Lic. en Ciencias de la Computación. Dentro de las asignaturas que conforman el Programa Analítico de esta carrera se encuentra Prolog, la cual es impartida en 4to año cuando los estudiantes hasta el momento han estado trabajando con un estilo de programación totalmente diferente. De allí que ellos manifiesten que esta asignatura constituye una de las más difíciles a las que se tienen que enfrentar durante toda la carrera.

Estudios realizados indicaron que resulta difícil que los estudiantes comprendan el funcionamiento de Prolog cabalmente y no existen novedosas herramientas en el proceso de enseñanza-aprendizaje capaces de cubrir esa necesidad al nivel de generalidad que se requiere. A esta conclusión se pudo arribar por la aplicación de diversas entrevistas a profesores y encuestas a un grupo de estudiantes donde se determinó que existen dificultades con la motivación de los estudiantes hacia la asignatura, existe falta de preparación para asumir el nuevo paradigma de programación, carencia de métodos novedosos que apoyen al profesor a la hora de impartir la docencia e insuficiencias en el cubrimiento de los materiales bibliográficos entre otros.

La Programación Lógica posee diversos temas, el diagnóstico realizado evidenció que donde se presentaron las mayores dificultades descritas anteriormente es precisamente en el tema correspondiente a las Listas. Los estudiantes manifestaron que este contenido constituye el más largo, monótono y difícil de la asignatura y posee gran importancia porque de su asimilación depende el éxito en la comprensión de los restantes temas, teniendo en cuenta que las Listas constituyen la base para el trabajo con este estilo de programación.

Lo antes expresado permitió arribar al siguiente **problema científico**:

¿Cómo contribuir a mejorar el proceso de aprendizaje del tema Listas en la asignatura Programación Lógica (Prolog) para estudiantes de la carrera de Ciencias de la Computación de la Universidad Central de Las Villas?

Numerosas son las búsquedas que se han realizado en Internet acerca de la temática de la enseñanza de la Programación Lógica y los resultados obtenidos son desalentadores con respecto a lo que se esperaba obtener.

Es cierto que existe mucha bibliografía en la red actualizada y relacionada con el tema. Se puede encontrar gran cantidad de información de numerosos autores donde se describe los antecedentes, origen y evolución, aplicaciones en diferentes ramas, actualidad y alcance que posee este paradigma de programación, ejemplo de estos materiales tenemos el trabajo realizado por el Dr. Daniel Gálvez Lio profesor de Prolog del Grupo de Inteligencia Artificial de la UCLV. y el artículo publicado por Cristina Loyo, investigadora y directora del Lania, y Olga Padrón, candidato a doctor por la Universidad de La Habana, titulado “La programación Lógica y el Prolog”.

Pero teniendo en cuenta que el aprendizaje visual ha demostrado ser un excelente recurso para mejorar la calidad del aprendizaje, sobre todo en comparación con los viejos y obsoletos métodos de la escuela del pasado, el objetivo de la búsqueda es

encontrar algún tipo de software educativo, que caracterizado por su novedad, brinde más claridad al estudiante a la hora de comprender los procesos y funciones que componen este estilo de programación; y que por tanto constituya un apoyo para el proceso de aprendizaje. Se han encontrado algunas herramientas informáticas que apoyan el proceso de enseñanza-aprendizaje de la Programación Lógica, como es el caso del Software Educativo Teaching, elaborado por el profesor Dr. Mateo Lezcano Brito de la Universidad Central de Las Villas, pero sus características, que se estudiarán en el desarrollo del presente trabajo, no permiten que se ajusten completamente a los objetivos que se traza la presente investigación. También la profesora Msc. Lydia Rosa Ríos Rodríguez, profesora del Centro Universitario de Sancti Spíritus, ha abordado esta temática para su investigación de doctorado y no ha podido acceder a ninguna herramienta.

Se han realizado consultas a través de correo electrónico, foros de discusión en Internet y eventos a determinados expertos en la materia, dígame profesores universitarios que investigan en esta misma línea, como Giraldo Valdés y Zenaida García, y otras personas que se han dedicado a trabajar con el Prolog y el conocimiento sobre lo que se pretende encontrar es muy pobre. En el mejor de los casos se encuentran algunos tutoriales que sirven de gran ayuda como materiales complementarios para lo que en un aula se imparte a los estudiantes, por ejemplo tenemos “Prácticas de Lógica Prolog” por Faraón Llorens Largo y Ma Jesús Castel de Haro de la Universidad de Alicante, “Programación Práctica en Prolog” por José E. Labra G. de la Universidad de Ovideo, pero no distan mucho de ser las mismas herramientas tradicionales que siempre han existido, con la única diferencia que ahora están en un formato electrónico con numerosos enlaces llamativos que indudablemente sí facilitan la interactividad y organización en el estudio, pero aún así, los problemas para entender la Programación Lógica siguen estando presentes.

El **objeto de investigación** es el proceso de enseñanza-aprendizaje de la Programación Lógica en la carrera de Licenciatura en Ciencias de la Computación y

su **campo de acción** es el aprendizaje del tema Listas en la Programación Lógica en los estudiantes de la carrera de Licenciatura en Ciencias de la Computación.

Con el análisis realizado hasta el momento y teniendo en cuenta las ventajas que actualmente brindan las TIC desde el punto de vista pedagógico se propuso el siguiente **objetivo general**:

Implementar un simulador para mejorar el aprendizaje del tema Listas en la asignatura Programación Lógica para los estudiantes de la carrera de Licenciatura en Ciencias de la Computación.

Las **preguntas científicas** que guiaron la investigación son las siguientes:

1. ¿Cuales serán los presupuestos teóricos metodológicos que justifican a los simuladores como herramientas que apoyan el proceso de aprendizaje?.
2. ¿Cómo se desarrolla actualmente el proceso de enseñanza-aprendizaje del tema Listas en la asignatura Prolog en la carrera de Lic en Ciencias de la Computación?
3. ¿Cómo diseñar el software educativo, en su clasificación de simulador, que facilitará el aprendizaje del tema Listas en la asignatura de Prolog?
4. ¿Cómo implementar el simulador que facilitará el aprendizaje del tema Listas en la asignatura de Prolog?.
5. ¿Cómo validar el simulador que facilitará el aprendizaje del tema Listas en la asignatura de Prolog?.

Las respuestas a estas interrogantes condujeron a la realización de las siguientes **tareas**:

1. Determinación de los presupuestos teóricos-metodológicos de los simuladores para el aprendizaje de Prolog.
2. Diagnóstico del proceso de enseñanza-aprendizaje del tema Listas en la asignatura Prolog en la carrera de Lic en Ciencias de la Computación.
3. Diseñar un simulador para el aprendizaje del tema Listas en la asignatura Prolog.
4. Implementar un simulador para el aprendizaje del tema Listas en la asignatura de Prolog.
5. Validar la pertinencia del simulador una vez terminado por el criterio de experto.

### **Población**

La población estuvo integrada por los 36 estudiantes que reciben actualmente la asignatura Programación Lógica en la carrera de Lic en Ciencias de la Computación de la facultad de Matemática, Física y Computación de la Universidad Central "Martha Abreu" de las Villas.

La **muestra** seleccionada coincide con la población declarada anteriormente.

Para la investigación se utilizaron varios métodos. Dentro de los **métodos teóricos** se aplicó el **análisis-síntesis**, la **inducción-deducción**, el **análisis histórico-lógico** y la **modelación**. Dentro de los **métodos empíricos** se utilizó la **observación**, la **revisión de documentos**, la **encuesta y entrevista** y el **criterio de experto** para estimar la validez, pertinencia y posible efectividad del simulador realizado.

El método **estadístico** que se utilizó fue del **matemático** el **cálculo porcentual** para el procesamiento de los resultados de entrevistas y encuestas posibilitando obtener el grado de concordancia de las opiniones y precisar su significado.

La **novedad científica** de esta investigación es el empleo de las TIC en el proceso de aprendizaje de la Programación Lógica ya que hasta el momento solo existían métodos tradicionales que frenaban un tanto el aprendizaje.

Su **significación práctica** está dada por el simulador obtenido que facilite el aprendizaje de la Programación Lógica, además de la posibilidad de publicarlo en Internet, como un nodo dentro de un mapa conceptual para el aprendizaje de esta materia, y que pueda estar accesible gratuitamente para todos los interesados.

El presente informe consta de tres capítulos. En el primero se hizo una revisión de la literatura donde se evidencia a grandes rasgos el origen y evolución que han tenido las TIC y como ellas pueden ayudar a resolver problemas que se presentan en el actual proceso de enseñanza-aprendizaje, particularizando en la problemática de la presente investigación.

En el segundo capítulo se realizó el diseño metodológico, se justificó cómo se seleccionó la población, la muestra y los diferentes métodos utilizados para la investigación. Se realizó una valoración de las diferentes herramientas que existen actualmente relacionadas con la propuesta, destacando los aspectos positivos y negativos que permiten incorporar o retirar del producto final.

En el capítulo tres se realizó el diseño del software, para ello se obtuvo los diferentes diagramas y requerimientos que integraron las múltiples facetas del desarrollo del mismo, se concluyó este capítulo con la validación del simulador por el criterio de experto.

**Capítulo1: Nuevo reto de las TIC y su impacto en la Programación Lógica.**  
**1.1 Las Tecnologías de la Información y la Comunicación**

Se puede denominar como Tecnologías de la Información y la Comunicación (TIC) el conjunto de procesos y productos derivados de las nuevas herramientas (hardware y software), soportes de la información y canales de comunicación relacionados con el almacenamiento, procesamiento y transmisión digitalizados de la información. [COL04]

Como ejemplo de TIC tenemos La televisión por cable, los discos de video, los satélites, el telefax, las redes de computadoras, el procesamiento de información por computadora, los interruptores digitales, las fibras ópticas, los láseres, la reproducción electroestática, la televisión de pantalla grande y alta definición, los teléfonos portátiles y los nuevos procedimientos de impresión, cuyos fundamentos son las telecomunicaciones, la informática y la tecnología audiovisual.

Las TIC es un fenómeno que ha invadido todos los sectores de la vida, desde el trabajo hasta el ocio, los procesos de enseñanza y aprendizaje que se realizan en los diferentes niveles de educación, la economía porque permiten generar riqueza a distancia y en red superando las fronteras geográficas y políticas. Han impuesto también un cambio en las relaciones laborales, económicas, culturales y sociales, y un cambio en la forma de pensar de los propios individuos.

Como otros aportes de las TIC tenemos:

- Fácil acceso a una inmensa fuente de información
- Proceso rápido y fiable de todo tipo de datos
- Canales de comunicación inmediata
- Capacidad de almacenamiento
- Automatización de trabajos
- Interactividad
- Digitalización de toda la información

Según el Dr. Pere Marques Graells, de la Universidad de Barcelona, España, a pesar de las magníficas credenciales que hacen de las TIC instrumentos altamente útiles para cualquier persona, y por supuesto imprescindibles para toda empresa, existen diversas circunstancias que dificultan su más amplia difusión entre todas las actividades y capas sociales: [SMS04]

### Frenos para la expansión de las TIC

- Problemas técnicos: Incompatibilidades entre sistemas, poca velocidad para navegar por internet y para algunas tareas (reconocimiento de voz...)
- Falta de formación: conocimientos teóricos y prácticos, aptitudes y actitudes favorables
- Problemas de seguridad: accesos no autorizados, inseguridad al realizar compras virtuales
- Barreras económicas: a pesar del progresivo abaratamiento el precio de los equipos es alto para muchos. Y la absolvencia exige nuevos gastos.
- Barreras culturales: el idioma inglés es el dominante.

### ***1.2 Impacto de las nuevas tecnologías en los sistemas educativos.***

A finales del siglo pasado, la casi totalidad de los campos profesionales han visto incrementado su potencial de desarrollo con la incorporación de las nuevas tecnologías y ello ha motivado un cambio sustancial en el modo de ejercer las funciones específicas en cada uno de estos campos. Esta situación lleva como contrapartida la demanda implícita de nuevos roles para el desarrollo profesional. En el caso concreto de la educación no hay excepción, a lo largo de estos últimos años, se enfatiza mucho en relación con el cambio en el perfil del maestro como consecuencia de la integración de las nuevas tecnologías en el ámbito escolar. [RIO05]

En el contexto de la sociedad actual y para cubrir sus expectativas se requiere elevar la calidad de la educación, en el sentido que entendemos este, en un proceso en el cual

no puede excluirse el uso de las TIC que no deben constituir simplemente un medio más, sino un recurso en el cual se sustenten las exigencias actuales para cambiar el mundo en el que educamos a niños y jóvenes. [MAT06]

En los últimos años ha surgido una serie de definiciones dentro del campo de la tecnología educativa. Muchas veces los conceptos de medios y tecnologías, se confunden. Los medios, por ejemplo el video, la computadora, el retroproyector, etc., son los productos usados dentro de un sistema de aprendizaje para lograr determinados objetivos. Sin embargo la denominada Tecnología Educativa la consideramos como una compleja organización de muchos elementos que están diseñados para ayudar a causar cambios en el comportamiento de los estudiantes al brindar la posibilidad, entre otras cosas, de una mayor atención a las diferencias individuales. [RIO05]

El creciente desarrollo de las nuevas tecnologías de la información ha conllevado a que los sistemas educacionales sufran transformaciones para adecuarse a una sociedad en estado de cambio permanente, con nuevos valores y necesidades.

Estos cambios, se concretan en:

1. Se desplaza el interés de la enseñanza hacia el aprendizaje.
2. El rol del profesor de expositor a guía y en última instancia como administrador de los medios.
3. Pasar de una cultura basada en el libro y en el texto a una cultura multimedios.
4. La desincronización de la educación (en el tiempo y en el espacio). (Todos podremos aprender en distintos momentos y en lugares diferentes. [RIO05])

Una computadora con un buen software puede producir un aprendizaje con calidad con respecto a ciertos procedimientos de trabajo, pero también un mal uso de este puede traer aparejado dificultades de aprendizaje muy serias para el estudiante. En ocasiones

hemos encontrado personas que se preguntan: ¿el uso de los medios y en particular de las computadoras originará automáticamente un aprendizaje efectivo?. Sencillamente la respuesta sin dudar es que NO. En este sentido, la forma en que se utilice, la motivación a proporcionar, el ajuste de las necesidades educativas, entre otras cuestiones, es lo que propiciará un mejor aprendizaje. [RIO05]

En la medida en que se adquiera una cultura informática, la sociedad estará en mejores condiciones de resolver sus problemas. La informática y la Educación no son un fin en si ni podemos ubicarlas fuera del contexto social. En este momento no se trata de analizar si las computadoras deben formar parte o no del proceso de enseñanza-aprendizaje, de disciplina o asignatura, eso una gran mayoría lo acepta, el problema está en ¿de qué forma nos puede y debe ayudar a enriquecer la labor educativa de los futuros profesionales que la sociedad necesita?

### ***1.3 La formación del profesional de la rama de la Informática.***

Evidentemente para poder adquirir y seguir desarrollando todas las nuevas tecnologías que disfrutamos actualmente se hace sumamente necesario la preparación de todo el personal profesional que las desarrolle y que sea capaz de darle seguimiento. Los profesionales que están vinculados con esta tarea son principalmente los Ingenieros Informáticos, los Cibernéticos, los Telecomunicadores, Automáticos entre otros. Dentro del plan de estudio de las carreras de Ingeniería Informática y la Lic en Ciencias de la Computación se encuentra la disciplina de Inteligencia Artificial. De los elementos que integran esta disciplina solamente una parte correspondiente a la asignatura Lógica Matemática estuvo presente desde los inicios de la especialidad. Con el transcurso del plan B, el desarrollo científico-técnico alcanzado en este campo exigió la modificación y ampliación de dichos elementos y como resultado de este proceso surge la asignatura Programación Lógica con el propósito de abarcar los elementos de la Inteligencia Artificial y de los lenguajes a ella asociados (en particular Prolog).

Como puede notarse, esta asignatura deberá jugar un papel significativo en la formación del profesional informático. La anterior afirmación se verifica con el análisis de cada una de las asignaturas como un sistema integrado.

La asignatura Prolog le permitirá al estudiante abordar problemas que dependan fuertemente de la experiencia, presentan elementos de incertidumbre, sean adecuados para su solución por métodos deductivos, inductivos u otros procesos cognitivos, y que son de difícil solución descriptivamente o de gran complejidad, aspectos estos que se pueden presentar en los trabajos que enfrenta un Ingeniero en Informática o Cibernético.

Con estos conocimientos y habilidades el egresado no sólo podrá abordar nuevas problemáticas, sino que también desarrollará sistemas informáticos mejor estructurados, más flexibles y potentes, y de mayor calidad profesional.

Actualmente esta materia se imparte en asignaturas de pregrado y de postgrado en Cuba y en el extranjero.

Los objetivos instructivos que presenta la asignatura de Prolog son los siguientes:

- Diseñar, implementar, administrar, mantener y controlar la calidad de sistemas que utilicen técnicas de Inteligencia Artificial (IA) para el control del funcionamiento y la toma de decisiones en organizaciones de base.
- Resolver creativamente problemas de IA ejecutando planes de acción pertinentes e innovadores.
- Garantizar la seguridad de los sistemas informáticos inteligentes.
- Integrar los sistemas informáticos inteligentes sobre plataformas diferentes.
- Evaluar, seleccionar y asesorar en la compra de ambientes de programación, medios técnicos y la asimilación de nuevas tecnologías a ser utilizados en las aplicaciones de IA.
- Preparar y entrenar al personal necesario para la explotación de los sistemas que utilicen las técnicas de IA.

### ***1.4 Algunos criterios sobre la Programación Lógica***

Una de las preocupaciones más tempranas de la computación de los años cincuenta fue la posibilidad de hacer programas que llevaran a cabo demostraciones automáticas de teoremas. Así empezaron los primeros trabajos de inteligencia artificial que más de veinte años después dieron lugar al primer lenguaje de programación que contempla, como parte del intérprete, los mecanismos de inferencia necesarios para la demostración automática. Este primer lenguaje está basado en el formalismo matemático de la Lógica de Primer Orden y ha dado inicio a un nuevo y activo campo de investigación entre las matemáticas y la computación que se ha denominado la Programación Lógica... [MAN00]. El nacimiento de Prolog se remonta al año 1971 en la Universidad de Marsella, Francia, en el seno de un grupo de investigación en el campo de la Inteligencia Artificial.

Prolog es un lenguaje declarativo e interpretado, en este tipo de lenguajes se representan los conocimientos sobre un determinado dominio y sus relaciones. A partir de ese conocimiento, se deducen las respuestas a las cuestiones planteadas, es decir se obtiene una inferencia. El dominio lo constituye un conjunto de objetos. El conocimiento se formaliza mediante un conjunto de relaciones que describen de forma simultánea las propiedades y sus interacciones.

Prolog está hecho para representar y utilizar el conocimiento que se tiene sobre un determinado dominio. Más exactamente, el dominio es un conjunto de objetos y el conocimiento se representa por un conjunto de relaciones que describen las propiedades de los objetos y sus interrelaciones. Un conjunto de reglas que describa estas propiedades y estas relaciones es un programa Prolog. Es además utilizado para resolver problemas que envuelven objetos y las relaciones entre ellos [ALL01]. Es uno de los lenguajes que más éxito ha tenido por tal razón son muchas las compañías de software que han creado sus propias versiones del mismo.

Se pretende buscar dentro de las TIC una herramienta que permita mejorar el proceso de aprendizaje de la asignatura Prolog. Esta constituye un nuevo estilo de programación al que el estudiante no se ha enfrentado y constituye una de las más difíciles de la carrera, precisamente por lo abstracta que resulta su explicación y entendimiento utilizando los métodos tradicionales . Dentro de todos los procesos que conforman la Programación Lógica, el presente trabajo está orientado específicamente a modelar el funcionamiento de las Listas, lo cual constituye un tema de importancia dentro de la asignatura.

### **1.5 Listas en la Programación Lógica**

Una lista es una secuencia ordenada de elementos que pueden tener cualquier longitud. Los elementos de una lista pueden ser cualquier tipo de términos (constantes, variables o estructuras), o incluso otra lista. Está formada por dos elementos: la cabeza (primer elemento de la lista, conocido como head) y la cola (resto de elementos de la lista, conocido como tail). Ambos elementos se consideran componentes de la estructura cuyo nombre es el punto “.”

Las listas son unas estructuras de datos muy comunes en la programación no numérica. El final de una lista se suele representar como una cola que contiene la lista vacía. La cabeza y la cola de una lista son componentes de una estructura cuyo nombre es “.” [BRA86]. Si es vacía se escribe como un átomo de Prolog, [ ].

Las listas se utilizan casi siempre para representar nuestros datos.

Debido al frecuente uso de este tipo de estructura, Prolog permite una notación abreviada para representar listas. Los elementos de la lista aparecen separados por comas, y la lista completa aparece entre corchetes. [TOR06]

**[a, b, c]** Lista con los tres elementos a, b y c.

**[ ]** Lista vacía.

En la definición de predicados se pueden usar además variables y un separador:

**[a | L]** Lista con el elemento a en la cabecera y el resto en la variable L (cola).

**[a, b | L]** Lista con los elementos a y b en la cabecera y el resto en la variable L (cola).

**[X | L]** Lista con el primer elemento instanciado en la variable X y el resto en la variable L (cola).

**[X, Y | L]** Lista con el primer elemento instanciado en la variable X, el segundo en Y y el resto en la variable L (cola).

Algunos de los problemas que se pueden resolver con las listas son los siguientes:

- La inserción y eliminación de elementos a una lista.
- El de la relación miembro, que se define como: “un elemento dado es miembro de una lista si está en su cabeza o está en el resto de la lista”.
- La concatenación: consiste en dado dos listas, devolver una tercera lista que sería la concatenación de las dos anteriores [BRA86].

Si se define una relación  $\text{conc}(A,B,C)$  que se verifique si C es la lista obtenida escribiendo los elementos de la lista B a continuación de los elementos de la lista A. Por ejemplo:  $\text{conc}([a,b],[c,d],C)$  se verifica si C es  $[a,b,c,d]$ .

- La intersección de dos listas, viene siendo como la intersección de dos conjuntos, donde dadas dos listas se devuelve otra que es la intersección de ambas.

### ***1.6 Reflexiones sobre los Software Educativos.***

El Software Educativo es cualquier programa computacional cuyas características estructurales y funcionales sirvan de apoyo al proceso de enseñar, aprender y administrar. Un concepto más restringido de Software Educativo lo define como aquel material de aprendizaje especialmente diseñado para ser utilizado con un computador en los procesos de enseñar y aprender. [MAR05]

Entre la variada gama de tipos de software se destacan los software en los cuales el rol esencial del computador es participar como herramienta; otros tipos serían aquellos en los cuales el computador juega un rol de alumno y el aprendiz se convierte en profesor del computador y para finalizar, existen aquellos software donde el rol preponderante del computador es de apoyo al aprendiz, como ocurre con los juegos educativos, software de ejercitación y práctica, tutoriales y de simulación. [IAJ04]

No obstante según esta definición, más basada en un criterio de finalidad que de funcionalidad, se excluyen del software educativo todos los programas de uso general en el mundo empresarial que también se utilizan en los centros educativos con funciones didácticas o instrumentales como por ejemplo: procesadores de textos, gestores de bases de datos, hojas de cálculo, editores gráficos... Estos programas, aunque puedan desarrollar una función didáctica, no han estado elaborados específicamente con esta finalidad.

Las características del software educativo, posibilitan la clasificación de los mismos en dependencia de la función que realizan dentro del proceso docente – educativo. Es usual encontrar en la literatura las clasificaciones siguientes: [MAR05]

- Tutoriales: constituye un programa especializado en la enseñanza de un dominio específico del conocimiento, apoyándose para ello en el diálogo con el estudiante, en la consolidación de un conjunto de aspectos esenciales que por su complejidad requieren de un nivel de abstracción que permita la representación adecuada del conocimiento.
- Entrenadores: Diseñado con el propósito de contribuir al desarrollo de una determinada habilidad, intelectual, manual, o motora, en el estudiante que lo utiliza por lo que profundizan en las dos fases finales del aprendizaje: aplicación y retroalimentación. Se parte de que los estudiantes cuentan con los conceptos y destrezas que van a practicar.
- Repasadores: Diseñado con el propósito de contribuir a repasar determinados contenidos estudiados con anterioridad.
- Evaluadores: para producir propiamente las evaluaciones y para aplicarlas.

- Simuladores: Se interactúa con un micro mundo en forma semejante a la que se tendría en una situación real para lograr el conocimiento. Aunque en la práctica este micro mundo puede resultar una simplificación del mundo real, el alumno resuelve problemas, aprende procedimientos, llega a entender las características de un fenómeno o aprende que acciones debe tomar en diferentes circunstancias.
- Libros electrónicos: Los libros electrónicos constituyen aplicaciones que hoy se están desarrollando con vistas a múltiples propósitos, y en particular, para el apoyo del proceso de enseñanza – aprendizaje.
- Juegos Instructivos: permiten aprender jugando. Proporcionan un medio ambiente para facilitar el aprendizaje. Proporciona al estudiante un reto entretenido con un componente instructivo.

Tipos de juegos:

- Aventuras.
- Tableros.
- Combate.
- Psicomotores y acertijos.
- Lógicos y patrones.
- Roles.
- Sistemas expertos: Constituyen una parte materializada de la inteligencia artificial. Representa las características asociadas con la inteligencia humana, entendimiento del lenguaje natural, aprendizaje, razonamiento, resolución de problemas, etc.

### Características esenciales de los programas educativos

Los programas educativos pueden tratar las diferentes materias (matemáticas, idiomas, geografía, dibujo...), de formas muy diversas (a partir de cuestionarios, facilitando una información estructurada a los alumnos, mediante la simulación de fenómenos...) y ofrecer un entorno de trabajo más o menos sensible a las circunstancias de los alumnos

y más o menos rico en posibilidades de interacción; pero todos comparten cinco características esenciales: [PSN01]

- Son materiales elaborados con una finalidad didáctica, como se desprende de la definición.
- Utilizan el ordenador como soporte en el que los alumnos realizan las actividades que ellos proponen.
- Son interactivos, contestan inmediatamente las acciones de los estudiantes y permiten un diálogo y un intercambio de informaciones entre el ordenador y los estudiantes.
- Individualizan el trabajo de los estudiantes, ya que se adaptan al ritmo de trabajo cada uno y pueden adaptar sus actividades según las actuaciones de los alumnos.
- Son fáciles de usar. Los conocimientos informáticos necesarios para utilizar la mayoría de estos programas son similares a los conocimientos de electrónica necesarios para usar un vídeo, es decir, son mínimos, aunque cada programa tiene unas reglas de funcionamiento que es necesario conocer.

El software educativo es útil porque apoya al alumno en su proceso de aprendizaje al contribuir a generar su propio conocimiento, reforzar aquello en lo que se sienta débil o simplemente a realizar un repaso a manera de estudio. Con el software educativo se atienden a los distintos tipos de inteligencia y se propicia el ser autodidacta en los estudiantes.

Debemos tomar en consideración que algunos de estos software están concebidos para ser empleados dentro de una actividad docente regular, orientada y dirigida por el profesor, mientras que otros están diseñados para ser empleados por el estudiante en su actividad independiente, después de recibir una orientación previa para su uso, o simplemente, para ser empleados en procesos de autoaprendizaje.

### **1.6.1 Otras consideraciones acerca de los simuladores.**

Hoy en día la Informática permite contar con una amplia variedad de programas que pueden ser empleados con múltiples enfoques como medio de enseñanza. Cada uno de estos programas tiene propósitos específicos, dirigidos a contribuir con el desarrollo de diferentes funciones del proceso docente.

Los softwares educativos, y en especial los simuladores, brindan la posibilidad de transformar la pedagogía, hacia una pedagogía más efectiva y emotiva que cambie, mejore y fortalezca el papel educador del profesor. Una tendencia actual es la difusión de simulaciones por Internet mediante la existencia de servidores a los que los se conectan los usuarios y ejecutan las mismas de manera remota.

Los simuladores son programas que tienen por objetivo proporcionar un entorno de aprendizaje abierto, basado en modelos reales. Reproducen fenómenos y leyes naturales, ofreciéndole al estudiante un entorno exploratorio que le permita llevar a cabo una actividad investigadora, manipulando determinados parámetros y comprobando las consecuencias de su actuación, llegando a entender las características del fenómeno. [MEN00]

Las simulaciones se organizan siguiendo los principios del aprendizaje por descubrimiento. Se utilizan en procesos físicos y biológicos, experimentos peligrosos y o costosos, en fenómenos que en la vida real se producen en tiempos excesivamente largos o breves, en los cuales el alumno tiene la posibilidad de descubrir y aplicar sus leyes, mediante la experimentación.

La educación está muy necesitada de buenas simulaciones lo que implica que haya que ser muy cuidadoso a la hora de prepararla. Hay que crear entornos interactivos lo más cercanos posible a la realidad, ricos en estímulos que faciliten alcanzar los objetivos pedagógicos propuestos, sobre todo cuando la experimentación no se pueda hacer realmente, pero sin perder de vista que nada haya más rico que la experimentación directa en el mundo real. [MEN00].

Por estas características de los simuladores mencionadas anteriormente es que se espera encontrar en ellos la herramienta idónea que apoye el proceso de aprendizaje del tema Listas en Prolog, basándonos para ello en las ideas expuestas anteriormente que Prolog es un lenguaje abstracto y que las simulaciones ofrecen gran claridad a la hora de comprender un determinado proceso.

Una vez se concluya con la implementación del simulador, el objetivo será insertarlo como un nodo dentro de un sistema de mapas conceptuales para apoyar el aprendizaje de la asignatura Prolog, realizado por la Msc. Lydia Rosa Ríos, y que ya ha dado sus primeros pasos y se soporta sobre la herramienta cmap tools (<http://cmap.ihmc.us>); y aunque esto último no es objetivo de la presente investigación si se considera necesario partir de esta idea para comenzar con el análisis de las tendencias y tecnologías en las que se debe sustentar la propuesta.

### ***1.7 Los Mapas Conceptuales para el aprendizaje de Prolog.***

Construir y compartir conocimientos, aprender significativamente, contextualizar el aprendizaje y aprender a aprender, son ideas sobre cuyo significado se ha reflexionado desde hace mucho tiempo y se ha intentado utilizar herramientas que permitan llevar a la práctica, estas aspiraciones. Para ello se han empleado diferentes técnicas y estrategias. Precisamente una de estas técnicas que ayudan a aprender son los Mapas Conceptuales (MC) [FEB02].

Los MC han constituido desde entonces, una herramienta de gran utilidad para profesores, investigadores de temas educativos, psicólogos, sociólogos y estudiantes en general, así como para otras áreas sobre todo cuando se necesita tratar con grandes volúmenes de información [GUT04].

La autora Lydia Rosa Ríos Rodríguez ha investigado y justificado las ventajas de realizar un Mapa Conceptual para apoyar el aprendizaje de la asignatura Programación Lógica. En su investigación plantea, y se concuerda con ella, que una de las técnicas que

ayudan a aprender son los mapas conceptuales, porque en ellos el conocimiento está organizado y representado en todos los niveles de abstracción, a la vez que proporcionan un resumen esquemático de lo aprendido. [RIO05]

Los tiempos que corren imponen a los educandos la necesidad de aprender a aprender, los mapas conceptuales son una importante herramienta en este sentido porque con su uso el alumno participa de forma activa, libre y creativa en su propio aprendizaje pues se ve obligado a buscar información para enriquecer sus contenidos, se ve en la necesidad de optar por determinados conceptos, seleccionar los que incluirá en el mapa y elegir la jerarquización de los mismos. Por otra parte, cuando el mapa conceptual se realiza en grupos, es posible compartir y negociar los significados aprendidos bajo el respeto a las opiniones de los demás y el rechazo a cualquier imposición no razonada.

Se puede decir que los MC tienen una gran importancia en el aprendizaje ya que [MAR01]:

- Facilitan una rápida visualización de los contenidos de aprendizaje.
- Favorecen el recuerdo y el aprendizaje de manera organizada y jerarquizada.
- Permiten una rápida detección de los conceptos clave de un tema, así como de las relaciones entre los mismos.
- Favorece el desarrollo del pensamiento lógico.
- Los materiales elaborados facilitan el estudio independiente.
- Sirven como un modelo para que los alumnos aprendan a elaborar mapas conceptuales de otros temas o contenidos de aprendizaje.
- Permiten que el alumno pueda explorar sus conocimientos previos acerca de un nuevo tema, así como para la integración de la nueva información que ha aprendido.
- Organizar los conocimientos a partir de las principales relaciones entre los conceptos.
- Favorece el trabajo colaborativo.

Los mapas conceptuales son muy útiles en la educación a distancia y, junto con sus ligas y recursos, son también accesibles como páginas de HTML. Por eso, los sistemas insertados en sus iconos deben ser fáciles de descargar y resulta más provechoso si

esta operación no es necesaria.

### **1.8 Técnicas analizadas. Análisis crítico**

Para dar cumplimiento al objetivo general del presente trabajo que es implementar un simulador es necesario, además de estudiar su fundamento teórico y metodológico realizado anteriormente, hacer un recorrido informativo que permita diagnosticar la existencia o no de herramientas que resuelvan el problema planteado. Del análisis profundo de sus ventajas y desventajas se justificará el por qué del lenguaje y softwares utilizados, y en caso de ser necesario presentar un nuevo método de trabajo.

Como resultado de una exhaustiva búsqueda en Internet se encontraron algunas herramientas que apoyan de alguna manera la comprensión del funcionamiento del Prolog. Analizaremos ahora con más detalle las más importantes teniendo en cuenta el aporte brindado para la concepción del simulador que se pretende construir, resaltando los aspectos positivos y negativos que justifiquen nuestra elección.

#### **1.8.1 Software Educativo Teaching.**

El Sistema para la Enseñanza del Prolog (SEP), desarrollado por el Grupo de Informática Educativa de la Universidad Central de Las Villas, integrado entre otros profesores por el Dr. Mateo Gerónimo Lezcano Brito está compuesto por los siguientes módulos:

1. Teaching: un sistema para la enseñanza del paradigma de la Programación Lógica.
2. Una metodología para programar sistemas expertos que utilizan el Prolog como máquina de inferencia.

3. Progen: un sistema que aplica la metodología anterior para la generación automática de sistemas expertos.

El sistema Teaching incluye un tutorial estructurado como hipertexto que explica teóricamente los conceptos fundamentales del lenguaje, a fin de que el estudiante pueda analizarlos tomando en cuenta su definición formal o más precisa. [MAT06]

El componente principal de Teaching es su herramienta de simulación, la cual dispone de un conjunto de programas ejemplos que facilitan el aprendizaje al mostrar los caminos de ejecución tomados y poder apreciar como influyen estos sobre cada una de las estructuras de datos utilizadas.

La idea medular del sistema Teaching, es enseñar a programar observando el comportamiento de ejemplos significativos que ilustran los mecanismos fundamentales del Prolog, mostrando como dichos mecanismos resuelven el problema planteado. De esta forma sencilla, el estudiante participa en la construcción del conocimiento ya que a partir de estos datos se generan algunas de las cláusulas Prolog que formarán el programa. [MAT06]

Teaching permite ejecutar paso a paso el programa, pudiendo apreciar los mecanismos típicos del lenguaje, tales como la unificación, el backtracking, la recursividad y otros. Entre los programas ejemplos que implementa el Teaching tenemos:

1. Programa para obtener las derivadas de una función.
2. Ubicar ocho reinas en un tablero de ajedrez sin que se ataquen mutuamente.
3. Torres de Hanoi.
4. El problema de programar un autómata de pila no determinístico. [MAT06]

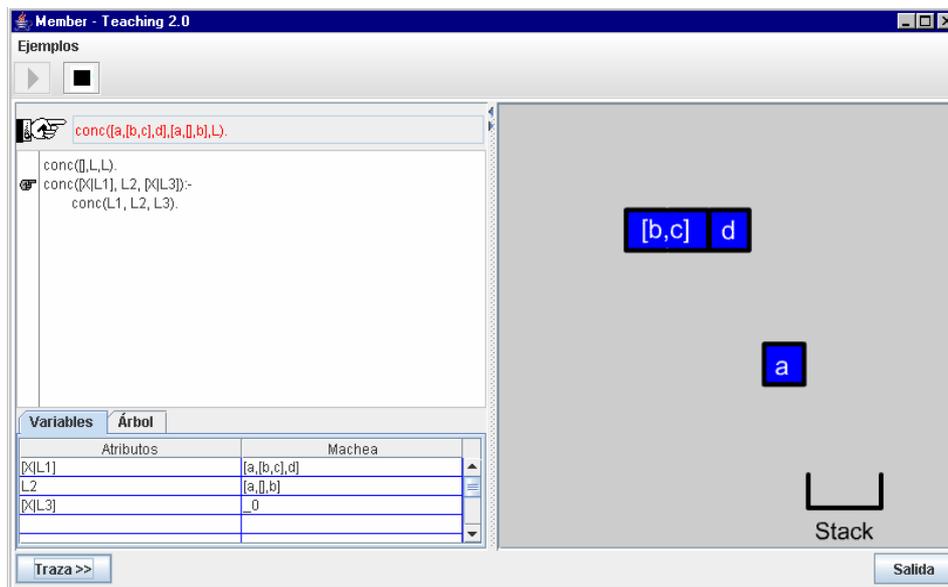
Este software está implementado en Arity Prolog con algunos elementos de Ensamblador [WEL06]. Y este es precisamente el problema fundamental que presenta esta herramienta por lo que no responde a las necesidades planteadas en esta

investigación. Se está tratando de implementar una interfaz más acorde con las necesidades actuales, que las que se podían brindar con el Arity Prolog. Además, no es bueno insertar los programas generados por el Teaching dentro de un mapa conceptual ya que, debido a su complejidad, dificulta la navegación por la red. Hay que descargar los archivos necesarios y no resulta cómodo para el estudiante, que puede perder el interés por las simulaciones o temer por la seguridad de su máquina. [WEL06]. Además en los nodos del mapa conceptual que se está desarrollando, se pretenden insertar programas más específicos o puntuales a lo que en determinado momento se está accediendo, o sea, este simulador se insertará específicamente en el nodo correspondiente al tema de Listas y el sistema Teaching no posee ninguna opción específica para este tema. Se prevé que el nuevo software presente ejercicios más sencillos o básicos que los que posee el Teaching, teniendo en cuenta que es para apoyar los primeros pasos del aprendizaje del tema.

Ante la dificultad mencionada anteriormente y tomando en cuenta que la nueva aplicación va a trabajar a través de la WEB, fue necesario buscar vías alternativas para resolver este problema y poder brindar al alumno una forma fácil y ágil de acceso a las simulaciones.

### **1.8.2 Teaching 2.0**

En La Universidad Central de Las Villas en el curso 2005-2006 se realizó un trabajo de diploma llamado “Un método para elaborar simulaciones del lenguaje Prolog” tutorado también por el Dr. Mateo Gerónimo Lezcano Brito. En esta tesis se obtuvo un sistema llamado Teaching 2.0, el cual constituye una nueva versión del sistema Teaching descrito anteriormente.



**Figura1 Sistema Teaching 2.0.**

Este sistema presenta una forma o método de trabajo que permite realizar simulaciones de los mecanismos internos dentro del lenguaje Prolog, para ello se simulan varios ejemplos de trabajos con listas como por ejemplo: el predicado miembro, la concatenación y la intersección. La materialización de este trabajo se aprecia a través de una aplicación en Java, la cual permite entre otras cosas, explicar de forma visual cómo se alcanzó un objetivo, cómo se instancian las variables, la animación de lo que ocurre en cada paso, así como el árbol de cláusulas obtenido. Del mismo modo se puede explicar por qué una conclusión particular no puede ser alcanzada. [WEL06].

Como se dijo anteriormente las simulaciones se construyen como aplicaciones de java, los desarrolladores justifican su elección por la amplia aceptación de Java por parte de la comunidad internacional de Internet y el hecho de que está soportada en diferentes plataformas de software.

Esta aplicación interpreta el código de Prolog y genera un árbol con las cláusulas que fueron accedidas y las que fallaron después de una consulta determinada.

Para la realización de estas simulaciones el problema es necesario correrlo en el software **WProlog**, cuyas características se estudiarán a continuación, (o en cualquier otro intérprete de Prolog) para conocer que línea del programa fue accedida en cada momento y el estado de las variables; sin esa información es imposible hacer las herramientas simuladas pues es utilizada para mostrar todo el proceso de ejecución de un programa Prolog. [WEL06].

Los predicados son implementados en el Amzi Prolog, idea tomada del proyecto Teaching.

Las animaciones obtenidas no son más que la representación visual y animada de estas trazas generadas en el WProlog, que se cargan y son únicas del problema que simulan y de la consulta que cada ejemplo resuelve. Pueden ser realizadas en Java, en Flash o en cualquier otro editor. Los desarrolladores utilizaron Flash.

### **1.8.3 Características de W-Prolog**

La Programación Lógica hoy en día ya se encuentra relacionada con Java. De esta relación aparecen una serie de aplicaciones de Programación lógica en Java entre las que se encuentra el interpretador de Prolog en Java (W-Prolog), creado por Michael Winikkoff. [MEN00]

W-Prolog es un interpretador implementado en JAVA para un lenguaje como Prolog. La implementación es extremadamente “portable” y puede ser ejecutada como un applet en un Browser. W-Prolog tiene una agradable interfaz para usuarios mejor que los sistemas prolog (los cuales proporcionan una interfaz típica con líneas de comandos). Está disponible tanto para Unix como para Windows, además puede ser ejecutado como un applet o como una aplicación individual, también puede ser llamado desde otro programa.

- La versión applet de W-Prolog tiene diferentes botones, figura 2:

Benchmark: Ejecuta el benchmark, la fórmula  $496/(\text{tiempo de ejecución})$  da la velocidad en interferencias lógicas por segundo.

About : Entrega información sobre el autor (Michael Winikkoff) y la versión (actualmente 1.2) dentro del área de texto del programa.

Append: Entrega un simple apéndice dentro del área de texto.

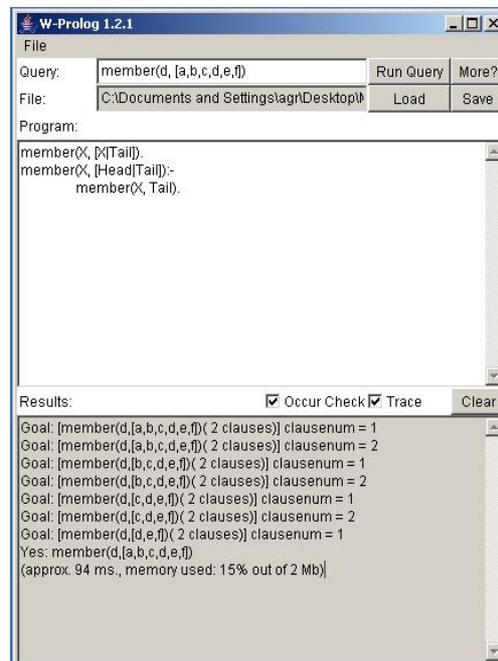
Documentation: Abre una ventana de navegador (Browser) en este archivo.

El botón del área de texto contiene la salida del W-Prolog, el botón “clear” limpia los resultados del área de texto .



**Figura2 Versión applet del W-PROLOG.**

La forma individual. Versiones anteriores de W-PROLOG apoyan la carga de archivos en la versión applet. Este tipo no tiene inconvenientes cuando se ejecuta con un visor de applet de SUN pero se rehusa a trabajar bajo netscape. [MAN00]



**Figura3 Versión de ejecución individual del W-PROLOG.**

Además del W-Prolog existen otros interpretadores de Prolog en Java como por ejemplo MINERVA, JAVALOG, JINNI, PROLOG CAFÉ, DGKS PROLOG E INTERPROLOG, los cuales no son objetivo para la investigación. [MAN00]

### Desventajas W-Prolog

El sistema W-Prolog es pequeño y es comparativamente simple. Sin embargo esto no lo hace particularmente rápido. Los applets tienen un número de limitaciones, por ejemplo, ellos no pueden escribir los archivos o abrir conexiones de redes. Como una consecuencia de esto los applets del W-Prolog no permiten cargar ni guardar programas.

### **1.8.4 El lenguaje Java.**

Aspectos positivos del lenguaje Java:

- Un solo código funciona para todos los browsers compatibles con Java o donde se tenga una Máquina Virtual de Java (Mac's, PC's, Sun's, etc).

- Un browser compatible con Java deberá ejecutar cualquier programa hecho en Java, esto ahorra a los usuarios tener que estar insertando "plug-ins" y demás programas que a veces quitan tiempo y espacio en disco.
- Permite desarrollar páginas Web dinámicas, se le pueden poner toda clase de elementos multimedia que permiten un alto nivel de interactividad, sin tener que gastar en paquetes carísimo de multimedia. [IAJ04]
- El JDK es una herramienta libre de licencias (sin costo), creada por Sun. Está respaldado por un gran número de proveedores. [PSN01]
- Debido a que Java nació en la era post-Internet, fue diseñado con la idea de la seguridad y la fiabilidad, por lo que se le integraron varias capas de seguridad para evitar que programas maliciosos pudiesen causar daños en los sistemas, sobre los que ejecuta la implantación de la Máquina Virtual Java. [MEN00].
- También se puede decir que el lenguaje está muy orientado al trabajo en red, soportando protocolos como TCP/IP, UDP, HTTP y FTP.

Según el criterio de algunos investigadores y programadores, obtenido de foros de discusión en Internet Java también tiene algunos inconvenientes que es válido resaltar:

- Hay diferentes tipos de soporte técnico para la misma herramienta, por lo que el análisis de la mejor opción se dificulta. [PSN01]
- Para manejo a bajo nivel deben usarse métodos nativos, lo que limita la portabilidad.
- El diseño de interfaces gráficas con awt y swing no es simple. [PSN01]

Se puede decir que ninguno de estos inconvenientes mencionados representa un problema para la calidad del producto final que se espera obtener, por lo que ellos no opacarían las ventajas encontradas a partir de las cuales se obtiene los resultados esperados para alcanzar el objetivo. Aún cuando Java proviene de una idea relativamente antigua, ha venido a revolucionar la computación de nuestros días,

debido a que integra tres características de suma importancia: la seguridad, la escalabilidad y la portabilidad.

Los **applets** de java son programas diseñados para ejecutarse como parte de una página web. Java impone restricciones de seguridad para que los applets no puedan dañar el ordenador en que se ejecutan. Como otras ventajas tenemos:

- El applet es independiente de la máquina y el sistema operativo. El programador no tiene que hacer ningún cambio en función de la plataforma en la que se ejecutan.
- Debido a las restricciones de seguridad del propio lenguaje, los errores de programación no alteran el sistema operativo del cliente.

Crear un sistema en otro lenguaje de programación distinto de Java no brindaría los requerimientos necesarios para navegar por la red. Queda claro entonces que los applets de Java constituyen el mejor candidato al reunir todas las condiciones que se necesitan para ubicar el simulador dentro de los nodos del mapa conceptual.

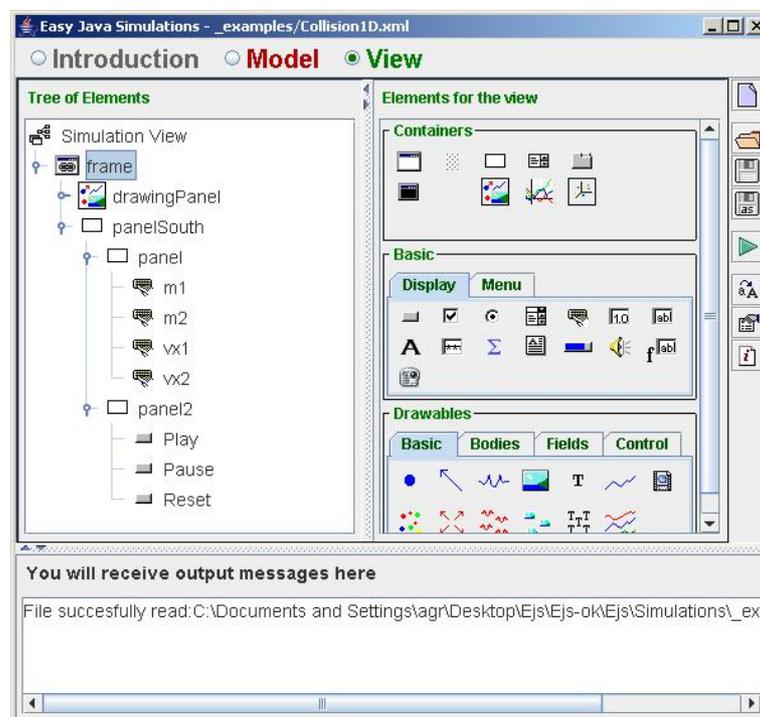
### **1.8.5 Easy Java Simulations**

Otro resultado obtenido en el estudio realizado fue un software conocido por sus siglas como Ejs lo que quiere decir Easy Java Simulations, creado por F. Esquembre, profesor del Departamento de Matemáticas de la Universidad de Murcia de España. [ESQ05]

Ejs, es un entorno gráfico de programación orientado al desarrollo de simulaciones científicas y técnicas en lenguaje Java. La construcción de estos applets Java puede ser sencilla o compleja, dependiendo del problema a resolver, se realizan a través de un entorno gráfico sencillo, básicamente con la utilización del Mouse, donde estudiantes y profesores del campo de las ciencias y de la ingeniería que poseen conocimientos básicos de programación pueden trabajar perfectamente.

Estos applets de Java creados son las simulaciones en sí, y pueden ser utilizadas como programas independientes bajo diferentes sistemas operativos, o se pueden distribuir de manera inmediata vía Internet y ejecutadas como applets dentro de páginas html utilizando algunos de los navegadores de Web más populares, debido a que el lenguaje de Java tiene como ventaja que los applets que genera pueden ser incluidos en los archivos html con las facilidades que ello implica en cuanto a ser aplicaciones multiplataforma y utilizados en Internet.

La siguiente figura muestra una vista del mismo:



*Figura4 Vista del Easy Java Simulations.*

En Ejs hay que describir un **modelo** del fenómeno que está bajo estudio, donde se definan e inicialicen las variables (que pueden corresponder a constantes, parámetros o variables de estado) que describen al sistema, y las relaciones entre ellas que se traduce en las leyes que gobiernan este fenómeno. Se tiene, además, un **control**, que define ciertas acciones que un usuario puede ejecutar sobre una simulación y, una

**vista**, que muestra una representación de los diferentes estados que el fenómeno puede tener, o sea visualiza continuamente el estado del modelo. [MAR05]

A pesar de que se pueden construir applets de cualquier tipo con esta herramienta, ella está principalmente diseñada para resolver problemas de Física, y crear simulaciones de otro tipo es particularmente complejo ya que definir variables y ecuaciones que controlen el movimiento de las imágenes dificulta el trabajo, por tanto esta herramienta no es la ideal para realizar el simulador propuesto.

### ***1.9 Algunas consideraciones acerca del NetBeans***

**NetBeans** es el programa de desarrollo para aplicaciones Java de licencia libre más utilizado por los usuarios. Este cuenta con las funcionalidades necesarias para el desarrollo de todos los tipos soportados bajo este estándar. El NetBeans IDE es un producto libre y gratuito sin restricciones de utilización.

Este IDE de desarrollo creado por Sun, el cual mantiene una licencia Open Source sobre Netbeans, ofrece ventajas extras sobre otros IDE's y llega a ser considerado el mejor editor de programas JAVA. Este software consta por el momento de dos partes, el IDE (entorno de desarrollo) NetBeans, y NetBeans Tools Platform (Plataforma de herramientas). Además de lo gratuito de la herramienta, esta posee un poderoso trasfondo de ejecución que permite ejecutar sin problemas aplicaciones de escritorio con la máquina virtual de Java y en aplicaciones Web, el servidor JSP Tomcat.

Todas las funciones del IDE son provistas por módulos. Cada módulo provee una función bien definida, tales como el soporte de Java, edición, o soporte para el sistema de control de versiones. NetBeans contiene todos los módulos necesarios para el desarrollo de aplicaciones Java en una sola descarga, permitiéndole al usuario comenzar a trabajar inmediatamente

---

## Capítulo 2: Propuesta de diseño del Simulador para el aprendizaje del tema Listas en Prolog.

### ***2.1 Justificación de la población y muestra.***

La población estuvo integrada por 36 estudiantes de 4to año de la carrera de Lic en Ciencias de la Computación de la UCLV, representando el 100% de la matrícula en ese año académico. Fueron seleccionados estos estudiantes teniendo en cuenta que eran los únicos que recibían la asignatura en el curso 2006-2007, período en el que se realizó la investigación.

Se seleccionó este centro de estudio para aplicar el instrumento por sus características y su prestigio actual. En ella laboran como profesores e investigadores profesionales de reconocida experiencia, tanto en Cuba como en el extranjero, en el tema de la Inteligencia Artificial que aportaron valiosos criterios a la investigación.

La carrera pertenece a la facultad de Matemática, Física y Computación, la cual fue fundada en 1955 ocupando actualmente un lugar destacado en la investigación y colaboración con el extranjero. Esta carrera alcanzó la condición de Carrera Certificada según el Sistema de Acreditación de Carrera del País en Marzo del 2004.

A partir del estudio a esta población se pudo realizar el diagnóstico de las causas del problema investigado. Se consideró oportuno no seleccionar ninguna muestra por las siguientes razones:

- el método utilizado para la validación del sistema fue el criterio de experto en el cual no es necesario la realización de análisis experimentales a muestras.
- fue muy difícil aislar a un grupo de estudiantes a los cuales se le aplicó el simulador como herramienta complementaria y que estos no divulgaran al resto de los estudiantes la nueva variante para el estudio.
- fue imposible encontrar homogeneidad dentro de un grupo de estudiantes en cuanto a su rendimiento docente debido a que en esto influyen

diversos factores como: motivación hacia la asignatura, regularidad en la carga docente e influencia de factores sociales y personales entre otros.

## **2.2 Métodos y técnicas de la investigación educativa.**

La única forma de estudiar un fenómeno sobre bases científicamente sustentadas y determinar su esencia es utilizando el método científico, el cual puede ser teórico y empírico.

En la concepción de esta investigación se utilizaron un sistema de métodos y técnicas de la investigación educativa con sus correspondientes instrumentos.

Los **métodos teóricos** fueron:

- Mediante el **análisis-síntesis** se pudo valorar la influencia que tienen los diferentes factores que intervienen en el proceso de enseñanza-aprendizaje, así como la relación existente entre esto y los problemas que existen actualmente para el aprendizaje de la asignatura de Prolog.
- La **inducción-deducción** permitió establecer generalizaciones en relación con los resultados científicos de la investigación a partir del análisis particular de los criterios de diferentes autores y de la teoría científica.
- El **análisis histórico-lógico** para estudiar a través de diferentes fuentes de información la evolución y la marcha progresiva de las teorías fundamentales sobre las cuales se realiza la presente investigación.
- La **modelación** para modelar las exigencias y características del simulador propuesto que permitirán al mismo una correcta reproducción de la realidad simplificada.

Los **métodos empíricos** se presentaron en cada área cumpliendo funciones de estudiar nexos, verificar características etc .

- Entre los empíricos se aplicó la **observación** del proceso de enseñanza-aprendizaje a la cual se enfrentan los estudiantes y profesores de la carrera de Lic en Ciencias de la Computación. Para ello se realizaron visitas a clases con el objetivo de analizar la manera en que el profesor impartía la docencia, si realizaba una motivación adecuada, si los métodos en los cuales se auxiliaba para la explicación de las Listas eran factibles, si contaba con bibliografía y ejercicios actualizados, si existía una correcta asimilación por parte de los estudiantes y si había una buena planificación del tiempo para la actividad. Se participó en algunas actividades de autoestudio de los estudiantes, a fin de observar si era correcta la forma en que los estudiantes organizaban su actividad, en qué medios se apoyaban, si existían estudiantes más preparados que pudieran apoyar al resto en el estudio, así como el tiempo que dedicaban a estudiar este tema Listas en Prolog con respecto al resto de las asignaturas.
- Mediante la **Revisión de documentos** se realizó el análisis de documentos correspondientes al procedo docente educativo que servirán de base para establecer criterios evaluativos con respecto al simulador. Se consultó el expediente de la asignatura de los últimos tres cursos, valorándose tanto los resultados docentes como la planificación de la materia a fin de encontrar las debilidades. Se consultó el Plan Analítico de la Carrera y de la asignatura. Se realizó un estudio de la documentación que recoge las metodologías y aspectos a tener en cuenta para la realización de la propuesta.
- Un método que se utilizó posteriormente fue el de **criterio de expertos**, en el cual la muestra de 7 expertos se estructuró a partir del criterio efectividad de la actividad profesional del experto. Se utilizó la Metodología de la Preferencia para validar la funcionalidad del software obtenido.
- También la **encuesta** a los estudiantes de 4to. año de la carrera Lic en Ciencias de la Computación de la UCLV con el objetivo de detectar los problemas que existían con el proceso de aprendizaje de la asignatura Programación Lógica y la **entrevista** a los profesores con el objetivo de

hacer una caracterización de la situación de la enseñanza del tema Listas en la asignatura Prolog.

### **2.3 Resultado de la aplicación de los instrumentos seleccionados ●**

El diagnóstico realizado permitió conocer las dificultades que tiene la asignatura Prolog y su tema Listas, para ello se aplicó una **entrevista** a los profesores (Ver Anexo 1) con el fin de detectar las insuficiencias en el proceso de enseñanza-aprendizaje de Prolog.

Fueron entrevistados 5 profesores universitarios de los cuales 3 son Máster en Ciencias y 2 son Doctores en el tema de Inteligencia Artificial. Todos poseen más de 6 años de experiencia impartiendo docencia y son investigadores del tema. La categoría docente mínima que poseen es Profesor Asistente. Como resultado de las entrevistas realizadas se pudo apreciar que los principales problemas que se presentan son:

- La mayoría de los profesores expresaron que existían problemas con la motivación de los estudiantes y declararon que aunque muchos estudios enfatizan que el aprendizaje de Prolog no resulta complejo si se aborda como primer lenguaje, la realidad es que las cosas no suceden así y los estudiantes se enfrentan a este paradigma después de haber trabajado por mucho tiempo con un estilo de programación imperativo, lo que trae como resultado que rechacen el nuevo paradigma y que lleguen, incluso, a cuestionarse su utilidad incidiendo negativamente en la motivación.
- Los resultados de la asignatura de manera general no son satisfactorios. Plantearon que las posibles causas están en que Prolog constituye un nuevo estilo de programación al que el estudiante hasta este momento no se ha enfrentado. Históricamente esta asignatura constituye una de las más difíciles a la que se enfrentan los estudiantes precisamente por lo abstracto que resulta su explicación y entendimiento. Estos resultados están siendo

influenciados por el hecho de que su aprendizaje se efectúa sin un balance correcto ente los componentes académicos, laboral e investigativo, de modo que dicho proceso se adapte a los requerimientos y posibilidades de cada estudiante.

- Todos los profesores coincidieron en que de todos los temas que conforman la asignatura el que presenta mayores dificultades con respecto a la comprensión y al tiempo dedicado para su planificación es el correspondiente al trabajo con las Listas. Señalan que este contenido es muy largo y difícil, necesitando mucho tiempo para impartirlo y ejercitarlo. Muchas veces el tiempo planificado para la clase no alcanza y por tanto tienen que dejar algunos ejercicios orientados para el autoestudio. Plantean que es necesario un adecuado aprendizaje del tema Listas debido a que estas constituyen la base para el resto de los contenidos que se impartirán en la asignatura.
- El total de los profesores utiliza como materiales complementarios los gráficos en el pizarrón. Manifiestan no tener a su alcance ninguna herramienta que sustituya esta técnica tradicional y concuerdan en que este método es deficiente, ya que necesitan dedicar demasiado tiempo a un mismo tema porque se dificulta el entendimiento.
- La mayoría de los profesores concuerda en que del total de los proyectos finales de la asignatura solamente una minoría presenta buena calidad y posibilidades reales de aplicación. Estos resultados están estrechamente vinculados con la motivación del estudiante y con las habilidades adquiridas durante la asignatura para la aplicación de los nuevos conocimientos. Esto da la medida que no se están cumpliendo cabalmente los objetivos de la asignatura y que definitivamente cuando el profesional egresa no aplica lo aprendido.
- Una parte significativa de los profesores utiliza las mismas conferencias de cursos anteriores. Estas conferencias están en formato digital publicadas en la red y los profesores no tienen la posibilidad de imprimirlas para los estudiantes por los problemas existentes con el papel y la tinta. Queda a responsabilidad de cada estudiante la gestión de sus propios materiales presentándose problemas con el autoestudio.

- La universidad no cuenta con el aseguramiento material bibliográfico necesario para cada estudiante, solamente existe una reducida cantidad de materiales en la biblioteca donde deben ser consultados.
- El fondo bibliográfico de la biblioteca universitaria necesita ampliarse pues los estudiantes refieren que hay ausencia de libros relacionados con el tema de Prolog no pudiendo solicitarlos en préstamo externo y en muchas ocasiones, cuando se trasladan hacia la biblioteca para consultarlos, ya están siendo utilizados los libros disponibles y el estudiante pierde su tiempo. En períodos de examen de la asignatura esta situación se agrava aún más.
- Hay consenso en que los estudiantes disponen de máquinas en los laboratorios en las que pueden dedicar suficiente tiempo para estudiar, de allí que sería ideal que pudieran realizar el autoestudio en este lugar. Los laboratorios laboran los 7 días de la semana las 24 horas y la relación alumno –máquina es de 2 alumnos por máquina.

### **Encuesta** a estudiantes (Ver Anexo 2)

Antes de iniciar la encuesta se realizó una breve explicación a los educandos sobre el objetivo general de la investigación, solicitándoles su cooperación. Se confeccionó la encuesta (ver Anexo 2) la cual tenía la intención de caracterizar el proceso aprendizaje de la asignatura Programación Lógica y de su tema Listas en específico.

La encuesta se aplicó con carácter anónimo. Contó con un total de 8 preguntas.

Al analizar los datos que se obtienen de la encuesta aplicada a los 36 estudiantes que conforman la muestra se tiene que:

- Antes de comenzar a recibir la asignatura sólo un estudiante se había interesado anteriormente por la Programación Lógica lo que representa el 2.7% de la población; 12 estudiantes manifestaron que habían oído sobre este tema pero que no lo habían abordado, representando el 33%, y el resto reconoció que Prolog era totalmente desconocido hasta el momento en que se lo presentaron en clases.

- 27 estudiantes manifestaron que Sí es difícil adaptarse al nuevo estilo de programación, lo que representa el 75% de la población y el resto de la población, 9 estudiantes el 25 %, van comprendiendo satisfactoriamente su funcionamiento.
- 35 estudiantes, el 97 % de la población, expresaron que de todos los temas que conforman la asignatura el que más trabajoso y difícil les resulta comprender es el relacionado con el trabajo con las Listas. Los estudiantes manifestaron que este contenido constituye el más largo y monótono, no existiendo una correcta orientación y planificación por parte del profesor de cómo estudiarlo. Muchas veces tienen que llevarse ejercicios de estudio independiente los cuales no logran vencer y por tanto se va afectada la secuencia de los contenidos que le siguen. Sólo un estudiante manifestó que No le es difícil comprender el funcionamiento de las Listas.
- 22 estudiantes, el 80 %, afirmaron que el cubrimiento material de la asignatura es Regular, y el resto 14 estudiantes, para un 20 %, lo clasificaron de Insuficiente, debido a que sólo disponen de las conferencias digitales publicadas en la red no pudiendo acceder a ninguna vía de impresión. Además no cuentan con bibliografía propia para estudiar y en muchos casos cuando se dirigían hacia la biblioteca ya se habían agotado los ejemplares existentes en ella.
- El 100% de los estudiantes calificaron de Insuficiente el método que utiliza el profesor en clases, debido a que necesita explicar varias veces un mismo ejercicio en el pizarrón agotando el tiempo planificado para la actividad.
- El 100 % de los estudiantes necesitaron Mucho del autoestudio en la asignatura para comprender el funcionamiento de las Listas. Apoyándose básicamente en ejercicios impresos o en otros en formato electrónico.
- El 100 % de los estudiantes afirmaron que un simulador serviría de gran ayuda para entender el funcionamiento del tema Listas en Prolog. En primer lugar expresaron que la forma de utilizar esta herramienta es más factible pues tienen posibilidades de acceso a las computadoras de la universidad los 7 días de la semana a cualquier hora. También apuntaron que una

simulación brindaría más claridad para comprender determinados procesos abstractos que al profesor le resulta difícil explicar en clases.

Resultado de la **entrevista** realizada al Msc. Lydia Rosa Ríos, cliente del sistema, para la obtención de los **requerimientos funcionales**. (Ver Anexo 3)

- Usuarios que interactuarán con el sistema.  
Según el cliente del producto el sistema está ideado para uso exclusivo de los estudiantes, o sea, para apoyar el proceso de aprendizaje en el autoestudio. Esto constituirá una herramienta complementaria para que ejerciten lo que se les imparte en el aula. El profesor no hará uso de esta herramienta para impartir la docencia.
- Nivel de autenticación de usuario.  
El sistema no necesita tener ningún nivel de autenticación de usuario, podrá ser usado por toda persona interesada en el tema Prolog. Además como el sistema de ejercicios que tendrá serán fijos, no existe la posibilidad que el usuario pueda hacer ninguna modificación y atente contra la integridad del mismo.
- Funciones que deben estar incorporadas en el sistema.  
Se pretende ante todo que el sistema cuente con una animación donde se le represente al estudiante la línea del ejercicio a la que se va accediendo en cada momento de la ejecución.

También se desea que se le muestre una explicación lo más detallada posible del ejercicio, en qué consiste el mismo, cómo se instancian las variables, por donde comienza la corrida, el historial de lo que va sucediendo, así como la condición de parada del mismo.

Se quiere dar la opción al estudiante para que corra el ejercicio paso a paso, puede detener la ejecución en curso y comenzarla nuevamente.

El enunciado del ejercicio debe estar visible en todo momento, el estudiante deberá tener la opción de acceder al menú de ejercicios cada vez que lo desee, de manera que puede interrumpir lo que está haciendo y seleccionar un nuevo ejercicio para comenzar a estudiarlo.

- Lugar donde se publicará el software.

El software se publicará como un nodo de un mapa conceptual que se está desarrollando para apoyar el aprendizaje de la asignatura Prolog. Este nodo corresponde específicamente al tema Listas. El mapa conceptual se hará público en Internet y se accederá como una página HTML. De allí que el simulador debe cumplir con ciertas restricciones como por ejemplo: su posible publicación como página html, fácil y rápido de descargar, de manera que permitan alcanzar este objetivo.

- Modificaciones que realizarán los estudiantes cuando interactúen con el.

Hasta el momento está concebido que el simulador sea estático, o sea, que existan un conjunto de ejercicios predefinidos a los que el estudiante pueda acceder para estudiar, no es objetivo que ellos puedan crear sus propias animaciones.

- En cuanto a la ayuda del sistema.

No se considera necesario agregar una ayuda al sistema. El diseño del simulador es muy sencillo y todas las funcionalidades están claras, los ejercicios son elementales, no obstante a ello cada ejercicio posee una explicación detallada que está visible durante todo el tiempo que el estudiante está trabajando con el.

- Apariencia gráfica a tomar en cuenta a la hora de diseñar la aplicación.

La apariencia del simulador debe respetar las normas que ya se han definido para la construcción del mapa conceptual donde será insertado. De manera que los usuarios no se pierdan a la hora de interactuar con él. Ya se han definido algunos colores como el azul en varias tonalidades y el gris y algunos logotipos que son necesarios respetar e incluir a fin de garantizar la

relación de pertenencia entre ambos. El tipo de letra y su tamaño también han sido definidos, se seleccionó un tipo de letra clara y legible. La concepción de su estructura debe ser realizada respetando determinados patrones de su concepción pedagógica.

- Otras funciones que debe tener el sistema.  
No es necesario agregar un botón de salida. La manera de salir del sistema será siguiendo la vía tradicional de abandonar una página web por el botón que aparece en el extremo superior derecho.

## ***2.4 Fundamento de las tendencias y metodologías en las que se sustenta la propuesta.***

### **2.4.1 Tendencias Pedagógicas.**

Está demostrado que la presencia y formación de adecuados motivos y conductas para el estudio garantizan que el alumno desarrolle la actividad con placer y aumente su motivación, manifestando interés por el aprendizaje, haciendo que el estudiante busque sus propias vías para el conocimiento, bajo la orientación del profesor.

Actualmente, la situación que se discute es la compatibilidad de las Tecnologías de la Información y las Comunicaciones con las distintas tendencias pedagógicas contemporáneas.

Para la realización de una investigación con fines pedagógicos se hace necesario fundamentar el aporte a través de las tendencias pedagógicas porque en ellas están contenidas las concepciones e ideas que en correspondencia con acciones adecuadas por su sistematización determinan, con sus múltiples variantes y alternativas de organización, que el proceso de enseñanza-aprendizaje resulte mucho más efectivo.

La esencia de la tendencia pedagógica del enfoque **histórico-cultural** centrada en Vygotsky es una concepción dirigida en lo fundamental a la enseñanza, facilitadora de un aprendizaje desarrollador, en dinámica interacción entre el sujeto y su entorno

social, de manera tal que se establece y desarrolla una acción sinérgica entre ambos, promotora del cambio cuanti-cualitativo del sujeto que aprende a punto de partida de la situación histórico cultural concreta del ambiente social donde él se desenvuelve. [NRR04]

Precisamente la importancia del trabajo de Vygotsky está dada por la valoración que hace de la influencia de lo histórico y lo social. El planteó que la actividad humana transcurre en un medio social, en activa interacción con otras personas, a través de variadas formas de colaboración y comunicación. Actualmente los autores lo clasifican como un constructivista, y en realidad es así, Vygotsky era fiel a su posición constructivista, pero a diferencia de los cognocitivistas anteriormente estudiados no dejó de reconocer que los significados provienen del contexto social externo y esto es precisamente lo que hace que su obra sea superior. La raíz de este planteamiento viene desde muchos años atrás cuando Marx y Engels plantearon que el hombre piensa según vive, de ninguna manera vive como piensa.

Según Vigotski lo que las personas pueden hacer con la ayuda de los otros puede ser, en cierto sentido, más indicativo de su desarrollo mental que lo que pueden hacer por sí solos. [MII05] 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). Esto último es uno de los conceptos esenciales en la obra de Vygotsky y no es otra cosa que la distancia entre el nivel real de desarrollo, determinado por la capacidad de resolver independientemente un problema, 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.

La determinación de esta zona permite caracterizar el desarrollo de forma prospectiva (lo que está en curso de maduración), lo cual permite trazar el futuro inmediato del estudiante, su estado evolutivo dinámico, reconstruir las líneas de su pasado y proyectarlas hacia el futuro. [GON02]

Para la construcción del simulador se parte principalmente del enfoque histórico cultural y la teoría de la actividad porque explican con claridad cómo el proceso de aprendizaje se debe convertir en el centro de atención, a partir del cual se proyecte el proceso pedagógico y esto se demuestra con los siguientes análisis:

- Los softwares educativos constituyen la base más sólida en la que el estudiante de hoy debe sustentarse para lograr su formación como futuro profesional, si tenemos en cuenta que la computadora es un poderoso instrumento que brinda la oportunidad de transformar la pedagogía, hacia una pedagogía más efectiva y emotiva que cambie, mejore y fortalezca el papel educador del profesor.
- El simulador debe contener un conjunto de actividades docentes por donde transiten los estudiantes para aspirar a niveles superiores de desempeño y ejecución, para lo cual el profesor debe diseñar las tareas a aplicar y ser sensible a los avances progresivos del estudiante.
- Lo más cercano al estudiante y lo que le está rodeando constantemente debido al perfil en que se está desarrollando son las computadoras y todas las nuevas tecnologías para la información, entonces sería muy factible utilizar lo disponible en su sistema de relaciones más cercano para propiciar interés y un mayor grado de participación e implicación personal en las tareas de aprendizaje.

Por lo anteriormente planteado se espera un éxito en el objetivo propuesto para la investigación, debido a que se garantiza un aprendizaje organizado que contribuye a facilitar el camino para llegar a la zona de desarrollo próximo al minimizar el esfuerzo para el entendimiento.

Por otra parte, en Cuba algunas de las ideas más generales de este enfoque, por su fundamento filosófico, constituyen principios instrumentados en el sistema. Los resultados logrados en el perfeccionamiento del proceso pedagógico en la Educación Superior cubana, en la calidad del aprendizaje y en la formación de la

personalidad del estudiante, han demostrado hasta el momento la efectividad y las perspectivas de este nuevo enfoque en el proceso pedagógico. El esfuerzo realizado en esta dirección ha sido además fructífero en la medida en que ha demostrado su condición de enfoque viable para asegurar el desarrollo de las potencialidades humanas.

#### **2.4.2 Fundamentación de la metodología utilizada.**

Cada día la producción de software busca adecuarse más a las necesidades del usuario, esto trae como consecuencia que aumente en tamaño y complejidad. Para lograr la productividad del software se necesita un proceso que integre las múltiples facetas del desarrollo del mismo. A semejanza con los planos que un arquitecto diseña como el esquema director a partir del cual se construirá un edificio, los diagramas UML suministran el modelo de referencia para elaborar un Plan Director de Iteraciones que define los hitos principales del proyecto y facilita a todos los agentes involucrados, la utilización de un lenguaje común para comprender y comunicar la estructura y la funcionalidad del sistema en construcción. [SEI04]

El Proceso Unificado (RUP) es un proceso de desarrollo de software (conjunto de actividades necesarias para transformar los requisitos de un usuario en un sistema de software). Es un marco de trabajo genérico que puede especializarse para una gran variedad de sistemas de software, para diferentes áreas de aplicación, diferentes tipos de organizaciones, diferentes niveles de aptitud y diferentes tamaños de proyectos.

El Proceso Unificado está basado en componentes. Utiliza el lenguaje unificado de modelado (UML) para preparar todos los esquemas de un sistema de software. De hecho, UML es una parte esencial de RUP, sus desarrollos fueron paralelos. No obstante los verdaderos aspectos definitorios del proceso unificado se resumen en tres fases claves: dirigido por casos de uso, centrado en la arquitectura, e iterativo e incremental. [NRR04]

La metodología RUP ha demostrado ser eficiente en la modelación de sistemas de información orientados a objetos. Propone un proceso de desarrollo incremental e

iterativo y su herramienta CASE se ha utilizado en el proceso de análisis y diseño del sistema. Para el análisis y el diseño de la aplicación se sigue una Metodología de desarrollo con tecnología orientada a objetos (RUP) que utiliza notación UML (Unified Modeling Language).

Cualquier tipo de proyecto (incluidos los pequeños, los basados en Web, aquellos fundamentales para un proyecto y los proyectos integrados) permiten obtener unos resultados más acordes con las previsiones gracias a la aplicación del proceso RUP. [SEI04]

### **2.4.3 Rational Rose**

Para apoyar el trabajo con esta metodología ha sido desarrollada por la Compañía norteamericana Rational Corporation la herramienta CASE (Computer Assisted Software Engineering) Rational Rose en el año 2000. Esta herramienta integra todos los elementos que propone la metodología para cubrir el ciclo de vida de un proyecto y es por ello que se ha utilizado para elaborar todos los diagramas incluidos en este documento.

### ***2.5 Descripción del método de trabajo.***

Como se ha visto ninguna de las herramientas analizadas en el capítulo anterior cumple totalmente con los requerimientos que debe tener el sistema. Algunas sólo funcionan como intérpretes del lenguaje Prolog, otras como el Ejs, a pesar de generar un Applet, tiene características específicas que interfieren en la creación del tipo de animaciones que se pretenden realizar. El software Teaching 2.0 se ajusta casi completamente a lo que se está buscando pero presenta un gran inconveniente y es que el es en sí una aplicación dura en Java y por tanto no es posible publicarlo dentro del nodo del mapa conceptual como página HTML, donde para poder cumplir con este objetivo es necesario que la aplicación sea un Applet en java.

De allí que tomando la idea medular del software Teaching 2.0 se construyera, utilizando la herramienta NetBeans 5.0, una aplicación similar en forma de applet.

A continuación se describen las principales tareas que se desarrollaron para obtener el nuevo simulador:

- Escoger e implementar el problema que se quiere simular después de un minucioso estudio acerca de en qué consiste, si es útil para el aprendizaje del estudiante, cuáles son los mecanismos de Prolog que utiliza y cuáles necesitan ser mostrados con más énfasis, si su implementación aporta elementos al desarrollo del alumno y su nivel de complejidad. Es necesario también crear una consulta adecuada.
- En lugar de realizar animaciones similares a las que aparecen en el Teaching 2.0 se consideró oportuno realizar una explicación detallada del ejercicio y lo que va ocurriendo en cada momento.
- En el Teaching 2.0 la clase TrazaFrame3 crea un objeto Prolog, cuya clase se define dentro del paquete del mismo nombre y contiene el programa escrito en Prolog y el resultado de la traza del WProlog; también contiene métodos para su creación y manipulación. En el método getProlog() de la clase TrazaFrame3, la cadena text debe ser llenada con el programa Prolog y la cadena goalStr, con el resultado que devuelve el WProlog después de ejecutarse el predicado con la consulta determinada. Esta consulta se le pasa como parámetro al constructor del objeto Prolog, así como el nombre del ejemplo. La clase PrologInformer es la interfase entre el objeto Prolog y la clase visual, ella se encarga de mostrar lo que este objeto contiene. [WEL06]. En la nueva aplicación fue necesario crear una nueva clase de tipo Applet, llamada NewJApplet e insertarle y modificarle el código descrito anteriormente, esta sería la clase principal del software y de esta manera se garantiza que el mismo sea un Applet.
- Se reutilizaron y modificaron otras clases como la Goal, Clause y Prolog.

## ***2.6 Análisis del diseño del sistema.***

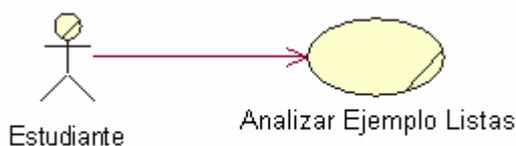
### **2.6.1 Diagrama de Casos de Uso del Negocio.**

El modelo del negocio describe el negocio en términos de casos de usos del negocio, que corresponde a lo que generalmente se le llama procesos.

El modelo de Casos de Uso del Negocio es un modelo que describe los procesos de un negocio (casos de uso del negocio) y su interacción con elementos externos (actores), tales como socios y clientes, es decir, describe las funciones que el

negocio pretende realizar y su objetivo básico es describir cómo el negocio es utilizado por sus clientes y socios. [GON02]

Un actor del negocio es cualquier individuo, grupo, entidad, organización, máquina o sistema de información externos; con los que el negocio interactúa. Lo que se modela como actor es el rol que se juega cuando se interactúa con el negocio para beneficiarse de sus resultados. En este caso el estudiante es el actor porque es el que interactúa con el negocio al analizar los ejercicios en clases. Obtiene beneficios porque adquiere conocimientos al comprender el funcionamiento de las listas.



*figura 5 Diagrama de Casos de Uso del Negocio*

### **2.6.2 Diagrama de Clases del Modelo de Objetos.**

El diagrama de clases, como artefacto que se construye para describir el modelo de objetos del negocio, muestra la participación de los trabajadores y entidades del negocio y la relación entre ellos [SEI04].

Un trabajador del negocio es una abstracción de una persona (o grupo de personas), una máquina o un sistema automatizado; que actúa en el negocio realizando una o varias actividades, interactuando con otros trabajadores del negocio y manipulando entidades del negocio. Representa un rol. En este caso el profesor es el trabajador del negocio porque explica a los estudiantes como funcionan las listas auxiliándose de sus diferentes métodos de enseñanza.

Las entidades de negocio representan a los objetos que los trabajadores del negocio toman, inspeccionan, manipulan, producen o utilizan durante la realización de los casos de uso de negocio. Comúnmente representan un documento o una parte esencial de un producto. Algunas veces representa cosas no tangibles como el conocimiento acerca de un mercado o cliente [SEI04].

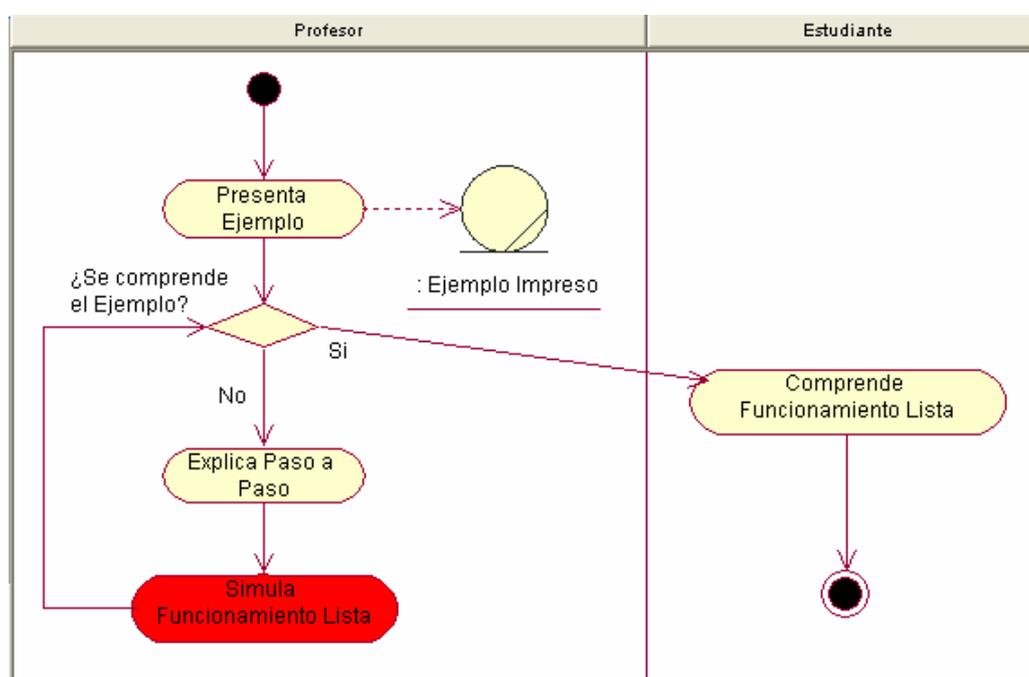


**figura 6 Diagrama de Clases del Modelo de Objeto**

### 2.6.3 Diagrama de Actividad.

Los diagramas de actividad se usan para mostrar cómo se construyen los diferentes flujos de trabajo o los procesos dentro de un sistema, cómo se inician, los variados caminos alternativos que se pueden tomar desde el inicio hasta el fin y dónde puede ocurrir el procesamiento paralelo durante la ejecución.

Un diagrama de actividades generalmente no modela el comportamiento exacto de un sistema de software (como lo hace un diagrama de secuencia), sino los procesos y los flujos a un muy alto nivel. Las actividades generalmente serán realizadas por uno o más casos de uso; la actividad describe el proceso que se desarrolla y tanto el caso de uso como un actor usará el sistema para realizar toda o parte de una actividad.



**figura 7 Diagrama de actividad**

Las actividades sombreadas son las que se automatizarán.

#### **2.6.4 Especificación de los requerimientos iniciales del software.**

Para determinar los requerimientos funcionales que debe tener el software fue necesario auxiliarse del criterio de un experto, en este caso se realizó una entrevista al cliente de este producto que es la Msc Lydia Rosa Ríos (Ver Anexo 3). De allí se pudo arribar a las siguientes conclusiones:

1. Mostrar un menú con el nombre de los ejercicios.
2. Organizar los ejercicios por orden de complejidad.
3. Mostrar una explicación lo más detallada posible del proceso que ocurre al dar solución al ejercicio.
4. Hasta el momento se han definido solamente cuatro ejercicios
  - 4.1 Verificar si un elemento es miembro de una lista.
  - 4.2 Concatenar dos listas
  - 4.3 Insertar un elemento dentro de una lista.
  - 4.4 Eliminar un elemento de una lista.
5. Realizar la simulación de los ejercicios.
  - 5.1 Ejecutar el ejercicio.
    - 5.1.1 Mostrar animación del proceso.
    - 5.1.2 Mostrar el historial del ejercicio.
    - 5.1.3 Mostrar valor de las variables.
  - 5.2 Detener ejecución del ejercicio.
  - 5.3 Mostrar marca que se terminó la ejecución.
6. Acceder a cualquier ejercicio en cualquier momento.
7. Salir del sistema.

#### **2.6.5 Requerimientos no funcionales.**

Los requerimientos no funcionales son propiedades o cualidades que el producto debe tener. Añaden funcionalidad al producto, pues hacen que sea fácil de usar, seguro, o interactivo demanda cierta cantidad de procesamiento. En muchos casos los requerimientos no funcionales son fundamentales en el éxito del producto debido a que forman una parte significativa de la especificación.

Apariencia o interfaz externa: El producto debe ser simple de usar, teniendo en cuenta las características de los usuarios hacia los cuales va dirigido. Debe ser

además autoritario para que los usuarios se sientan confiados al utilizarlo puesto que la información que se maneja es sumamente delicada.

Portabilidad: Se garantiza que la implementación del producto sea compatible con diferentes plataformas.

Usabilidad: Se prevé que la usabilidad de este producto sea bastante elevada, o sea, que cuente con elevado nivel de aceptación para los usuarios, debido a que constituye un poderoso instrumento para apoyar el aprendizaje de la asignatura Prolog. Además estará publicado en Internet.

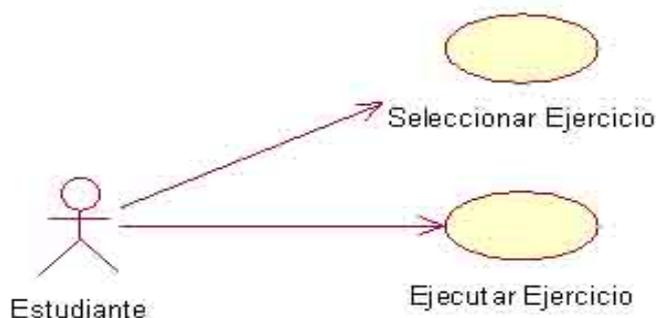
Rendimiento: En cuanto al rendimiento la mayoría de las funcionalidades que están previstas necesitan de una respuesta instantánea, por lo que se tendrá que garantizar la no ocurrencia de demoras con la selección de la tecnología realizada.

Requerimientos Legales: Este producto goza de entera legalidad puesto que no constituye ninguna violación a leyes al utilizarse para su confección tecnología de tipo software libre.

#### **2.6.6 Diagrama de casos de uso del sistema.**

Los casos de uso son artefactos narrativos que describen, bajo la forma de acciones y reacciones, el comportamiento del sistema desde el punto de vista del usuario. Por lo tanto, establece un acuerdo entre clientes y desarrolladores sobre las condiciones y posibilidades (requisitos) que debe cumplir el sistema [GON02]. Un diagrama de casos de uso representa gráficamente a los procesos y su interacción con los actores.

El actor es el ente que interactúa con el sistema, en este caso se ha definido solamente un actor que es el estudiante.



*Figura8 Diagrama de casos de uso del sistema*

### **2.6.7 Descripción de los casos de uso del sistema.**

Caso de Uso	Seleccionar Ejercicio
<b>Actores:</b>	Estudiante
<b>Propósito:</b>	Leer el nombre de los ejercicios que se van a simular.
<b>Resumen:</b>	Se selecciona el menú Ejemplo y se muestra el listado de los ejercicios disponibles.

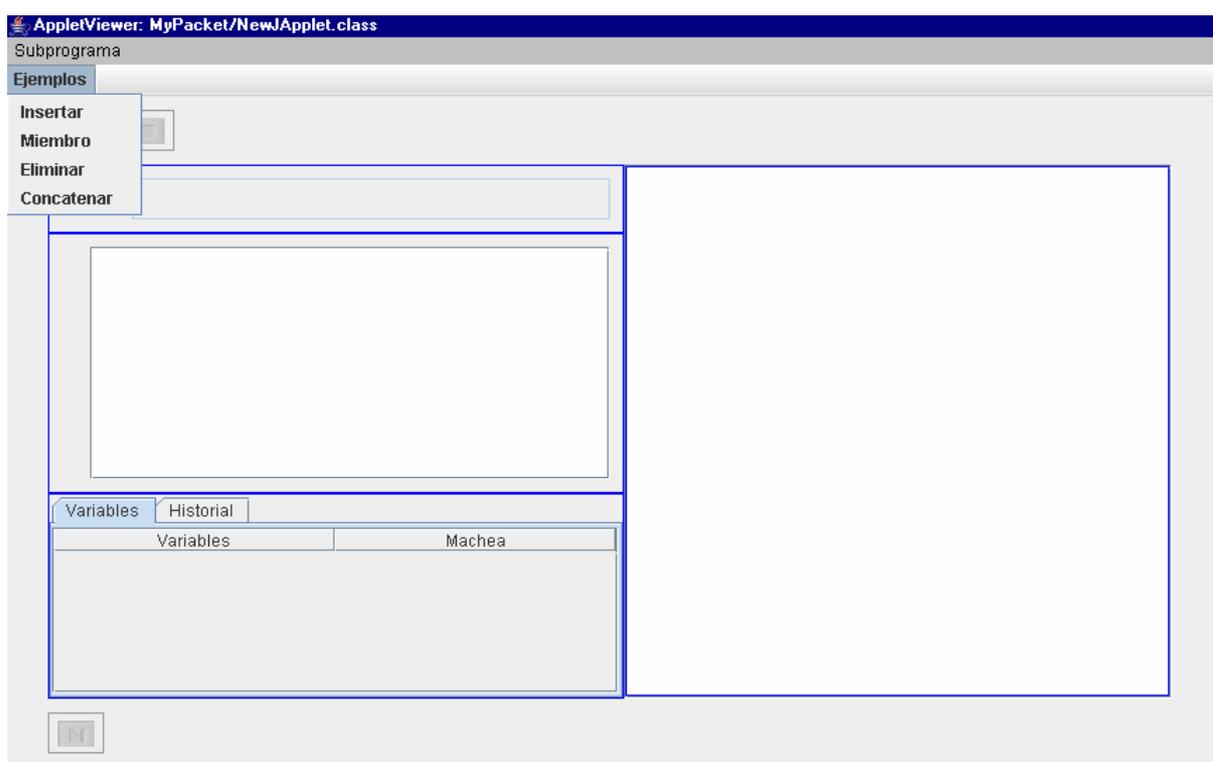
Caso de Uso	Ejecutar Ejercicio
<b>Actores:</b>	Estudiante
<b>Propósito:</b>	Ver la simulación de cada ejercicio.
<b>Resumen:</b>	Realizar la simulación de un ejercicio, mostrando la animación del proceso, las variables, el historial y su explicación.

### **2.6.8 Diagrama de navegación.**

Los diagramas o mapas parecen recursos adecuados para facilitar la "navegación" por la información. Se trata de una representación esquemática del espacio en el que se mueve el usuario, incluso señalándole dónde se encuentra en ese momento y la posibilidad de saltar directamente a otras secciones. Los diagramas o mapas que incluyen sólo grupos de nodos fuertemente relacionados simplifican los mapas (evitando la sobrecarga cognitiva) y facilitan percepciones globales de la estructura general.



**Figura 9 Mapa de navegación del sistema**



**Figura 10 Vista Menú Ejemplos**

Vista correspondiente al menú de Ejemplos donde se muestra un listado con todos los ejercicios que están simulados en el software.

AppletViewer: MyPacket/NewApplet.class

Subprograma

Ejemplos

`conc([a,b,c],[d,e,f],L)`

```

conc([],L,L).
conc([X|L1], L2, [X|L3]):-
  conc(L1, L2, L3).

```

Variables		Machea	
X		b	
L1		[c]	
L2		[d,e,f]	
L3		_288	

Obtener una lista que sea el resultado de la concatenación de L1=[a,b,c] y L2=[d,e,f].

R/

La Concatenación consiste en obtener una nueva lista (L3) que sea el resultado de adicionar L2 a continuación de L1.

El programa comienza probando la primera regla, al no encontrar como primer parámetro una lista vacía entonces pasa a instanciar con la segunda regla, obteniéndose como resultado el valor de L1 y L2, los cuales son los parámetros de la siguiente llamada recursiva a la regla uno. Debido a ello vemos repetidas las cláusulas dos veces.

Estos pasos se irán repitiendo hasta que L1 tome valor de cadena vacía matcheando con la primera regla y por tanto finalizándose la ejecución.

En la variable X se va guardando el valor en cada momento que tiene la cabeza de la lista L1.

L3 es una variable anónima, o sea, esta variable durante la corrida del programa prolog no tiene valor, de allí los números aleatorios que va tomando. Esto es debido a que prolog toma temporalmente estos valores intermedios y los pasa a una pila para finalmente por el mecanismo de la recursividad obtener en esta variable el resultado final del problema.

**Figura 11 Vista Simulación**

Este es un ejemplo de una simulación, específicamente la concatenación de dos listas. Se muestra a la derecha la explicación y a la izquierda el ejercicio en Prolog con las variables instanciadas y el historial.

### **2.6.9 Diagrama de interacción.**

Los diagramas de interacción no son más que una descripción del modo en el que cada operación detectada en los diagramas de secuencia lleva a cabo sus responsabilidades y modifica el estado del sistema. En UML los diagramas de interacción pueden representarse a través de los Diagramas de Colaboración y/o de los Diagramas de Secuencia.

El tipo de diagrama seleccionado para construir los diagramas de interacción fue el de Secuencia, debido a que muestra cómo los objetos se comunican unos con otros en una secuencia de tiempo, qué sucede en cada momento, y para ello contienen objetos con sus ciclos de vida y los mensajes que se envían entre ellos ordenados secuencialmente. [ALV00].

El diagrama de secuencias es el núcleo de un modelo dinámico, y muestra todos los cursos alternos que pueden tomar los casos de uso. Los diagramas de secuencias

se componen de 4 elementos que son: el curso de acción, los objetos, los mensajes y los métodos.

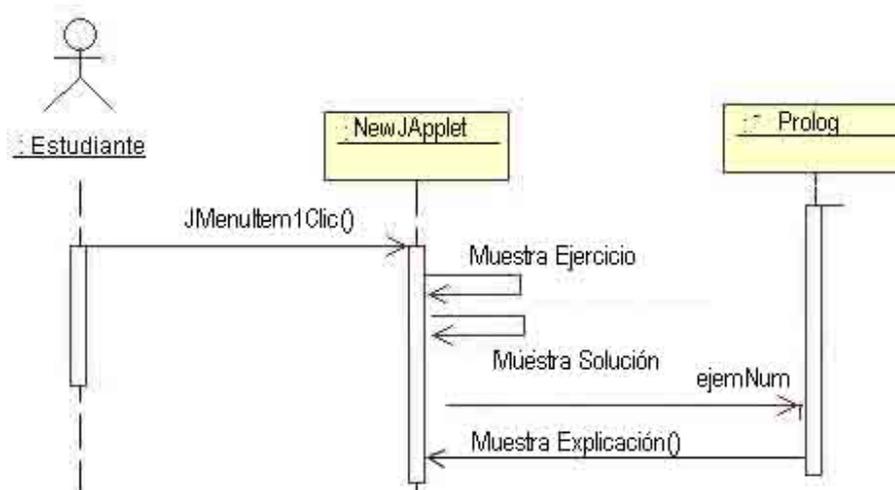


Figura 12 Diagrama de Interacción: Seleccionar Ejercicio

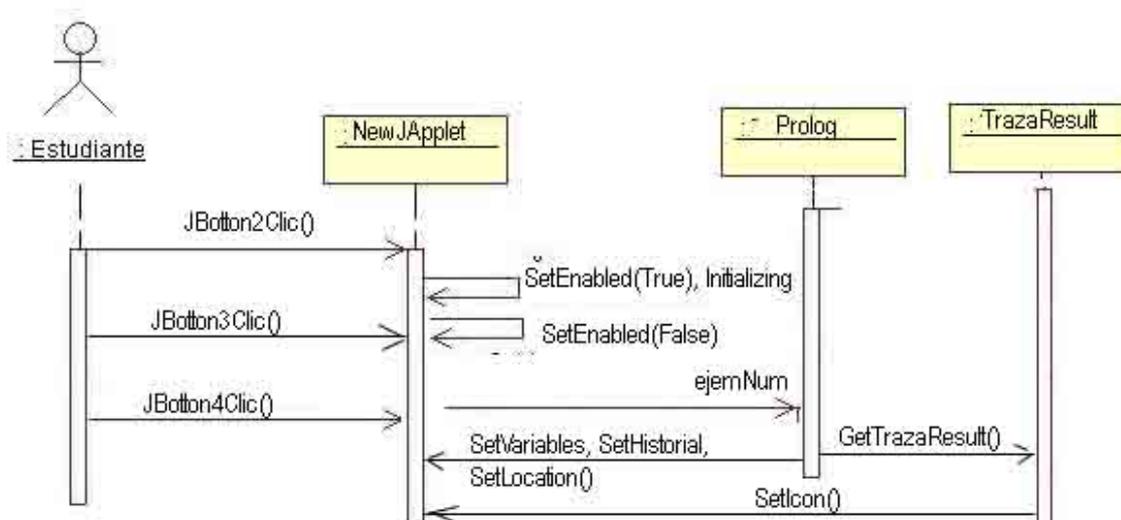


Figura 13 Diagrama de Interacción: Ejecutar Ejercicio

## 2.7 Valoración de la factibilidad del Simulador para la enseñanza del tema Listas en la asignatura de Programación Lógica por el criterio de expertos.

Para comprobar la validez del Simulador para la enseñanza del tema Listas en la asignatura de Programación Lógica se utilizó el método de validación por criterios de expertos.

La muestra de expertos se estructuró a partir del criterio: **efectividad de la actividad profesional del experto**. Como resultado se seleccionó a un grupo de 7 especialistas de reconocida experiencia profesional (Ver Anexo 4) avalada por:

- alta calificación científica y docente ( 7 especialistas para un 100%),
- conocimiento profundo de la informática (7 especialistas para un 100%)
- conocimiento del tema Prolog y la Inteligencia Artificial (5 especialistas para un 71.4%)
- resultados satisfactorios en el trabajo como profesores de la Educación Superior (100%).

Con esta muestra de expertos estuvieron representados 2 centros de educación superior, la Universidad Central “Marta Abreu” de Las Villas (UCLV), y el Centro Universitario “José Martí Pérez” Sancti Spíritus (CUSS). En la UCLV se seleccionaron a profesores del grupo de Inteligencia Artificial, así como a profesores y directivos de la carrera Licenciatura en Ciencia de la Computación. En el CUSS estuvieron representados a profesores investigadores de la facultad de Ingeniería.

Para la aplicación del método criterios de expertos se siguió la **Metodología de la Preferencia**. Para ello los expertos evaluaron el producto a partir de una guía elaborada (Ver Anexo 5) con las temáticas que se consideraron indicativas y medibles de la validez del producto. A continuación se reflejan los criterios emitidos por cada una de estas temáticas.

#### ***Aplicación en la educación superior.***

- Presenta alto nivel de aplicabilidad en la realidad escolar del Ingeniero Informático puesto que es un recurso a la altura de su perfil profesional y soluciona un gran problema latente al apoyar el aprendizaje del tema Listas en la asignatura Prolog, materia indispensable en su formación.

- Posee un alto nivel de aplicación porque permite a los profesores apoyarse en el uso de softwares educativos que propicien un aprendizaje más efectivo e independiente por parte del estudiante.
- Presenta un elevado nivel de aplicabilidad pues estará publicado gratuitamente en Internet, la vía más novedosa de compartir información en la actualidad.

### ***Necesidad del simulador como propuesta.***

- Es una imperante necesidad para solucionar varios de los problemas que afronta actualmente la asignatura Prolog.
- Es una necesidad debido a que actualmente existe mucho desinterés por parte de los profesionales de la rama de la informática en aplicar este estilo de programación a la solución de problemas y esta herramienta fomentará el interés en los alumnos al mostrarle claramente su aplicación a través de diferentes ejemplos.
- Es objetivamente necesaria la introducción de esta propuesta, pues asegura que los docentes desarrollen el autoaprendizaje.
- La temática que se aborda es compleja y por tanto resulta de mucho valor una ayuda de este tipo para guiar, orientar y apoyar la asimilación de estos contenidos.

### ***Nivel científico y actualidad.***

- Posee nivel científico y actualidad al abordar el aprendizaje de la Programación Lógica como aspecto medular, cuestión en la que hoy se presentan insuficiencias, en los alumnos y en la forma de proceder de los profesores, que interfieren en el proceso de enseñanza – aprendizaje.
- Tiene actualidad en el contexto educacional actual puesto que es una herramienta de tipo software educativo que se corresponde con los enfoques actuales de la educación superior cubana
- El nivel científico lo corrobora todo el proceso seguido desde la exploración del estado actual del problema hasta las estrategias y procedimientos que se sugieren.

- Aborda un aspecto de gran actualidad en las carreras que imparten este tema en nuestras universidades al brindar estrategias para conducir el aprendizaje de Prolog donde el alumno participa activamente.

### ***Fiabilidad funcional***

- El software se ejecuta en condiciones normales y no muestra mensajes error.
- Se adapta a cualquier configuración de pantalla, navegador que se utilice y sistema operativo.
- Todos los ejercicios estarán habilitados de manera que se puede interrumpir la ejecución de uno para comenzar con otro.
- Es posible detener la ejecución en curso en cualquier momento y comenzarla nuevamente.
- El enunciado y explicación detallada del ejercicio estará siempre visible para el estudiante.

### ***Comunicación e Interfaz.***

- La interfaz del software es amigable, cumple con los estándares web establecidos: accesibilidad, usabilidad y de fácil manejo.
- Los medios empleados están armónicamente distribuidos, sin sobrecargar la pantalla.
- Los colores utilizados están en correspondencia con la interfaz del mapa conceptual donde estará insertado.

Los criterios de los expertos han sido expresados después del estudio de los tres capítulos del presente trabajo. A pesar de ser diversos los argumentos hay consenso en que la propuesta presenta posibilidades reales de aplicación para los alumnos y la consideran una necesidad insoslayable para perfeccionar la labor del docente en función de la formación de un estudiante que pueda comprender el funcionamiento de las listas en Prolog mediante la incorporación de un software educativo como una novedosa estrategia para apoyar el aprendizaje.

### **Conclusiones**

El estudio realizado en la presente investigación permitió detectar las insuficiencias que se presentan en el proceso de aprendizaje del tema Listas en la asignatura Programación Lógica dados fundamentalmente por la falta de motivación por parte de los estudiantes hacia la asignatura, la falta de preparación para asumir el nuevo paradigma de programación, carencia de métodos novedosos que apoyen al profesor a la hora de impartir la docencia e insuficiencias en el aseguramiento de los materiales bibliográficos.

Se propuso como solución confeccionar un software educativo específicamente un simulador que apoye el proceso de aprendizaje del tema Listas en Prolog, caracterizado por tres ejercicios sencillos que conjuntamente con el ambiente fácil y amigable de su diseño servirán de gran ayuda al estudiante para mejorar el aprendizaje.

La validación del simulador por medio del criterio de experto corroboró su validez, hubo consenso en la pertinencia y factibilidad de la propuesta para perfeccionar el proceso de aprendizaje de los alumnos.

### **Recomendaciones**

1. Se recomienda a la tutora de este trabajo publicar el software dentro del Mapa Conceptual como ha estado previsto hasta el momento.
2. Continuar profundizando en la concepción teórica del Simulador para el aprendizaje de las Listas en la Asignatura de Programación Lógica, a fin de seguir desarrollándolo mediante la inserción de nuevos ejemplos clásicos.
3. Utilizar el software como material complementario en la asignatura de Programación Lógica.
4. Divulgar los resultados de esta investigación mediante publicaciones y presentaciones en artículos y eventos científicos.

### Referencias Bibliográficas

**[TER96]** Terceiro, J. (1996) "Sociedad Digital". Madrid, Alianza editorial .

**[MII05]** Material Introdutorio al tema II en la Multimedia para el Diplomado La Educación en la Sociedad de la Información y el Conocimiento, 05

**[VAL00]** Valdés, G. (2000). "Algunas consideraciones metodológicas relativas a la elaboración de software educativo". Conferencia Magistral en el IV Taller Internacional sobre la enseñanza de la matemática en la ingeniería y la arquitectura. La Habana.

**[RIO05]** Ríos, R Lydia (2005). Trabajo Final de la Asignatura: Problemas Sociales de la Ciencia y la Tecnología en opción al título en Doctor en Ciencias.

**[LOY06]** Loyo, C; Padrón O. "La Programacion Lógica y el prolog", URL: [www.lania.mx/index.html](http://www.lania.mx/index.html), Visitada 05/03/2006

**[COL04]** Colectivo de autores, Elección de la tecnología, URL: [http://www.itlp.edu.mx/publica/tutoriales/produccion1/tema3\\_3.htm](http://www.itlp.edu.mx/publica/tutoriales/produccion1/tema3_3.htm), visitada:12/04/04.

**[MAR01]** Martínez Rina, "Aprender a Aprender II: Mapas Conceptuales", URL: <http://www.sindominio.net/aluned/bolened/>, visitada: Mayo del 2001.

**[SMS04]** Sun Microsystems Soluciones, URL: <http://es.sun.com/soluciones/pdf/Technicalespanol3.pdf>, visitado 21/01/04.]

**[JOR05]** Jordi Adell . "La navegación hipertextual en el World-Wide Web: implicaciones para el diseño de materiales educativos", URL: <http://tecnologiaedu.us.es/revistaslibros/a7.htm>, visitado: 22/12/05

**[EDW89]** Edwards, D.M. and Hardman, L. (1989). 'Lost in Hyperspace': Cognitive Mapping and Navigational in a Hypertext Environment. McAleese, R. (Ed.) Hypertext: Theory into Practice. Intellect Press: Oxford, págs. 104-125)

**[MAT06]** Mateo Lezcano Brito, Victor Giraldo Valdes Pardo . "Algunas Experiencias en la Utilización de Sistemas de EAC para la Enseñanza de la Inteligencia Artificial", URL: <http://www.emis.de/journals/DM/v62/art8.pdf>, visitado: 20/09/06

**[WEL06]** Welkis Pérez Carrillo, Andrés Gutiérrez Rodríguez, "Un método para elaborar simulaciones del lenguaje Prolog", Trabajo de Diploma, UCLV-2006.

**[ESQ05]** ESQUEMBRE, F., (2005) Easy Java Simulations. The Manual. Version 3.4, URL: [http://www.um.es/fem/Ejs/Ejs\\_es/index.html](http://www.um.es/fem/Ejs/Ejs_es/index.html), visitado: 20/09/06

**[MAR05]** Martín, S. Dormido, R. Pastor, J. Sánchez: Sistema De Levitación Magnética: Un Laboratorio Virtual En “Easy Java Simulation, 2005”

**[IAJ04]** Introducción a Java,  
URL: <http://www.monografias.com/trabajos/java/java.shtml>, visitado 14/05/04.

**[PSN01]** Primer Seminario Nacional de Elaboración de guiones de Software Educativo para a Escuela Cubana. La Habana. 2001.

**[MEN00]** Rolando Menchaca Méndez , Dr. Félix García Carballeira: “Arquitectura de la Máquina Virtual Java” \Mtro., 1 de Octubre de 2000 Vol.1 No.2, [http:](http://)

**[BAR99]** BARRERA, M. I., (1999) *Informática Avanzada. Aplicación de Programación Lógica en Java*. Valdivia.

**[MAN00]** Manuel Iván Barrera : “Aplicación de Programación Lógica en Java”, 2000

**[ALL01]** Allaire, Jeremy. “Desarrollo de Aplicaciones Dinámicas de Internet con Macromedia MX”, 2001.

**[BRA86]** BRATKO, I., (1986) *Prolog Programming for Artificial Intelligence*. London. Addison-Wesley.

**[SEI04]** Software Engineering Institute, Calidad, URL: <http://www.ne.com.co/html/esp/calidad.html>, visitado: 14/05/04.

**[NRR04]** “Lo nuevo de Rational Rose 2000” URL: [www.abists.com.mf/Fabs/Rational/notasTK](http://www.abists.com.mf/Fabs/Rational/notasTK) visitado: 17/02/2004

**[GON02]** González, Anaisa. Modelamiento del Negocio. Centro de Estudios de Ingeniería de Sistemas (CEIS), Cujae, 2002.

**[GON01]** González, Anaisa. Modelamiento del Sistema. Centro de Estudios de Ingeniería de Sistemas (CEIS), Cujae, 2001.

**[ALV00]** Álvarez, Sofía, González Anaisa. Metodología para el desarrollo de aplicaciones con tecnología Orientada a Objetos utilizando notación UML. La Habana, 2000

**[GOL99]** Goleman, D. (1998) La Inteligencia Emocional en la Empresa, Vergara, Argentina, 1999.

**[TOR06]** Torres de Alba, Roberto. Minimanualillo de SWI-Prolog, 22 de marzo de 2006.

**[FEB02]** Febles Juan P., Estrada Vivian, "Aplicaciones de la Inteligencia Artificial" Primera Edición, Parte III, Mapas Conceptuales, pag. 400, Ed. Universidad de Guadalajara, México, 2002.

**[GUT04]** Gutiérrez Enrique Javier "Los Mapas Conceptuales", Universidad de León. URL: <http://www3.unileon.es/dp/ado/ENRIQUE/Didactic/Mapas.htm> , visitado Abril, 2004

**[VEG97]** Vega Fernando José, "Mapas Conceptuales de Ontoria", URL: <http://venus.javeriana.edu.co/qualitas/>, visitado Mayo de 1997.

**[NOV98]** Novak, J.D. y Gowin, D.B. -Aprendiendo a Aprender.-. Ediciones Martínez Roca, S.A., Barcelona, 1988.

**Bibliografía**

1. Alfonso Sánchez, Ileana y otros. (2004). Tendencias pedagógicas contemporáneas. [on line] (accedido el 12 de septiembre de 2006). Disponible en: <http://www.sanmartin.edu.co/academicos>.
2. Álvarez de Zayas, Carlos. Didáctica. La Escuela en la vida. La Habana: Editorial Pueblo y Educación, 1999 p. 24.
3. Belloch, C. Teorías de aprendizaje y diseños instruccionales .En Informática y aprendizaje. [on line]. (accedido el 12 de septiembre de 2006). Disponible en: <http://cfv.uv.es/belloch/2tie4c11.htm>
4. Bravo Romero, Silvia. (2004). “El Mapa Conceptual como estrategia de enseñanza y aprendizaje en la resolución de problemas”. Facultad de Química. Universidad de la Habana.
5. Conociendo algo de software educativo. MODULO 3. Centro Zonal Sur - Area Pedagógica. [en línea]. (accedido el 16 de agosto de 2006). Disponible en: <http://www.enlaces.udec.cl/centrozonal-sur/pedagogica/Modulo3.pdf>.
6. Gómez, Luz del Carmen; Moncayo, José de Jesús. La Importancia de Promover en el Aula Estrategias de Aprendizaje para elevar el Nivel Académico en los Estudiantes. [on line]. (accedido el 20 de diciembre de 2006). Disponible en: <http://www.congreso.unam.mx/ponsemloc/ponencias/1224.html>
7. González, Y. (2004). Perspectivas de la “educación para los medios” en la escuela de la sociedad de la comunicación No 24. [en línea]. (accedido el 10 de marzo de 2006). Disponible en <http://www.campus-oei.org/revista/rie24a04.htm>.
8. Hernández Sampier, Roberto. (2003). Metodología de la investigación 1. Editorial Félix Varela. La Habana.
9. Hernández Sampier, Roberto. (2003). Metodología de la investigación 2. Editorial Félix Varela. La Habana.
10. Herrera, René. La informatización de la sociedad: un reto para la educación cubana. [on line]. (accedido el 1 de marzo de 2006). Disponible en: <http://www.somece.org.mx/memorias/2000/docs/453.DOC>.

11. Jacobson, Ivar, Booch, Grady. (2000). El proceso Unificado de desarrollo de Software. PEARSON EDUCACIÓN, S, A. Madrid.
12. Lenguaje de programación Java. [en línea]. (accedido el 11 de noviembre de 2006). Disponible en:  
[http://es.wikipedia.org/wiki/Lenguaje\\_de\\_programaci%C3%B3n\\_Java](http://es.wikipedia.org/wiki/Lenguaje_de_programaci%C3%B3n_Java)
13. Lino, Troadio; Alfonso, Ileana Tendencias pedagógicas contemporáneas [on line]. (accedido el 2 de marzo de 2006). Disponible en:  
<http://www.monografias.com/trabajos6/tenpe/tenpe3.shtml>
14. Marquès, Pere. Multimedia educativo: clasificación, funciones, ventajas e inconvenientes, 1999 <http://dewey.uab.es/pmarques/funcion.htm>
15. Martín Maglio, Federico. (2004). Teorías psicológicas del aprendizaje. [on line]. (accedido el 1 de marzo de 2006). Disponible en:  
<http://www.fmmeduccion.com.ar/Pedagogia/teoriaspsicoapren.htm>
16. Mesa A. Magda (2007). El Criterio de Experto. Reflexiones para su uso en la investigación. ISCF "Manuel Fajardo"
17. Meza M.,Mónica. (2000). La exploración del software como apoyo al aprendizaje.  
<http://centros5.pntic.mec.es/ies.marques.de.santillana/matem/softw.htm>
18. Primer Seminario Nacional de Elaboración de Guiones de Software Educativos para la Escuela Cubana. [on line]. (accedido el 2 de marzo de 2006). Disponible en:  
<http://bdigital.edusol.rimed.cu/textos/informatica/seminarioguionistas/SeminarioElabGuiones.doc>
19. Programa Analítico Carrera Ingeniería Informática.
20. Ríos Rodríguez, Lidia. (2004). Por qué los mapas conceptuales para representar conocimientos en softwares educativos.

21. Romo, Abel. El enfoque sociocultural del aprendizaje de Vygotsky [on line]. (accedido el 3 de marzo de 2006). Disponible en: <http://www.monografias.com/trabajos10/enso/enso.shtml>
22. SALCEDO, P. "Ingeniería de software educativo, teorías y metodologías que la sustentan". Revista Informática. edición N° 6. 2000 [on line]. (accedido el 12 de septiembre de 2006). Disponible en: <http://www.inf.udec.cl/revista/edicion6/psalcedo.htm>
23. SOLÉ, Isabel. Orientación educativa e intervención psicopedagógica. Barcelona: Editorial Horsori, 1998. 255p.
24. Tendencias Pedagógicas Contemporáneas. Colombia: Corporación Universitaria de Ibagué, 1994. 177p.
25. Tuckler, Eddy. Temas generales de Prolog [on line]. (accedido el 4 de marzo de 2006). Disponible en: <http://www.monografias.com/trabajos5/prolog/prolog.shtml>
26. Ugaz, Max. Lo nuevo de rational rose 2000. [en línea]. (accedido el 11 de septiembre del 2006). Disponible en <http://www.usmp.edu.pe/publicaciones/boletin/fia/info36/proyectos.html>
27. VALDES PARDO, Víctor Giraldo, Jacinto González Acosta y Emilio García Roselló. Consideraciones sobre el desarrollo sistemático del software educativo. 134 p. En: Ingeniería de Sistemas Educativos apoyados en Tecnologías [SEPAD]
28. Vilalta, Josep. (2001). Desarrollo de aplicaciones WEB con UML. [en línea]. (accedido el 6 de septiembre de 2006). Disponible en [http://www.umlpatterns.com/TRAD\\_obert/TRAD\\_WAE\\_abierto.pdf](http://www.umlpatterns.com/TRAD_obert/TRAD_WAE_abierto.pdf).

## **Anexo 1**

Entrevista a profesores que han impartido la asignatura de Prolog.

**Objetivo:** Caracterizar la situación actual de la enseñanza de la asignatura Prolog y del tema Listas.

Preguntas:

1. ¿Qué Categoría Científica o Título Universitario usted posee?
2. ¿Qué Categoría Docente o de Investigación posee?
3. ¿Qué Tiempo lleva impartiendo la asignatura?
4. ¿Qué tipos de materiales orienta para el estudio independiente?
5. ¿Qué vías utilizan los alumnos para acceder a la bibliografía necesaria?
6. ¿Cómo cataloga la motivación de los estudiantes por su asignatura?
7. ¿Cuál considera usted es el tema más difícil que tiene la asignatura y por qué?
8. ¿Cuál es el tema más importante de la asignatura y por qué?
9. ¿Cómo son los resultados académicos de los estudiantes?
10. ¿Qué métodos complementarios utiliza en la clase para apoyarse?
11. ¿Cómo cataloga la calidad de los proyectos finales de la asignatura?

## Anexo 2

Encuesta a estudiantes que están recibiendo la asignatura Prolog.

**Objetivo:** Detectar los problemas que existen con el proceso de aprendizaje de la asignatura Programación Lógica y en específico del tema Listas.

MARQUE CON UNA "X"					
Año que cursa: 1 ----- 2 ----- 3----- 4 ----- 5 -----					
Rango de edad					
19-21 (1)	22-25 (2)	26-29 (3)	30-33 (4)	34-37 (5)	37-40 (6)
1- ¿Te has interesado por el tema de la programación lógica anteriormente?					
a) Si ___ b) Más o menos ___ c) Nunca ___					
1.1 ¿Te resulta difícil este nuevo estilo de programación?					
a) Si ___ b) No ___					
2- ¿Te resulta difícil comprender el funcionamiento de las listas?					
a) Si ___ b) No ___					
3- La orientación acerca de cómo estudiar este tema es:					
Excelente	Muy bien	Bien	Regular	Insuficiente	
4- El cubrimiento material de las asignaturas es:					
Excelente	Muy bien	Bien	Regular	Insuficiente	
5- El método que utiliza el profesor para explicar en clases es:					
Excelente	Muy bien	Bien	Regular	Insuficiente	
6- En que medida necesitas del autoestudio para aprender el funcionamiento de las listas					
Mucho	Mas o menos	Poco	Muy Poco	Nada	
7- Para tu autoestudio en que te apoyas					
Impresos	Formato electrónico	Videos	Software	Otros	
8- Considera usted que un simulador le apoyaría en su aprendizaje de las listas en Prolog. ¿Por qué?					

### **Anexo 3**

Entrevista al cliente del sistema Msc. Lydia Rosa Ríos.

**Cargo:** J'Dpto Informática del CUSS

**Experiencia:** Máster en Informática Aplicada, 5 años como profesora del Centro Universitario, investigadora del tema de Prolog para su tesis doctoral.

**Objetivo:** Obtención de los requerimientos iniciales del software.

Preguntas:

P1: ¿Quiénes serán los que interactuarán con el sistema?

P2: ¿Tiene previsto algún nivel de autenticación de usuario?

P3: ¿Qué funciones deben estar incorporadas en el sistema?

P4: ¿Dónde se publicará el software?

P5: ¿Qué modificaciones van a realizar los estudiantes cuando interactúen con él?

P6: ¿Desea agregar alguna ayuda del sistema?

P7: ¿Qué apariencia gráfica sugiere usted se tome en cuenta a la hora de diseñar la aplicación?

P8: ¿Especifique todas las funciones que debe tener el sistema además de lo anterior expuesto?

## Anexo 4

Tabla que contiene los datos de los expertos que evaluaron la propuesta.

**Tabla: Características de los especialistas consultados.**

Nombre y Apellidos	Entidad	Cargo	Años de experiencia.
1. Msc. Lydia Rosa Ríos	CUSS	Jefe de Departamento de Informática y Profesor Asistente.	6 años
2. Msc. Aurelio Hernández Reyes	CUSS	Profesor Asistente	33 años
3. Msc. Vladimir Caballero	CUSS	Profesor Asistente Administrador Red CUSS	6 años
4. Msc Jorge Fardales.	CUSS	Profesor Auxiliar Coordinador de la Msc. NTE	13 años
5. Dra. Zenaida García.	UCLV	Profesora Titular	25 años
6. Dr. Mateo Lezcano	UCLV	Profesor Titular J'Grupo de Inteligencia Artificial	30 años
7. Dr. José Ignacio	CUSS	Profesor Auxiliar	23 años

## Anexo 5

Guía para la validación de la propuesta por criterios de expertos.

Compañero (a): Usted ha sido seleccionado por su experiencia en la enseñanza superior, como dirigente, como profesor de la carrera de Lic. en Ciencias de la Computación y por el nivel docente metodológico que posee, para que dé sus valoraciones sobre la propuesta: “Simulador para la enseñanza de las listas en la asignatura de Programación Lógica ”.

### I. Datos Generales:

Nombres y Apellidos:

---

Centro de trabajo:

---

Años de experiencia en educación: \_\_\_\_\_ años.

Como docente: \_\_\_\_\_ años.

Como informático \_\_\_\_\_ años.

Como dirigente \_\_\_\_\_ años.

II. Sobre la propuesta usted podrá expresar su criterio teniendo en cuenta los siguientes aspectos:

1. Aplicación en la educación superior.
2. Necesidad del simulador como propuesta.
3. Nivel científico y actualidad.
4. Fiabilidad Funcional.
5. Nivel de comunicación.

Gracias por su colaboración.

