



**CENTRO UNIVERSITARIO
DE SANCTI SPÍRITUS
“JOSÉ MARTÍ PÉREZ”**

TESIS EN OPCION AL TITULO DE MASTER EN NUEVAS TECNOLOGÍAS PARA LA
EDUCACION

*Título: Simulador para el aprendizaje del
proceso de matching*

Autor: Ing. Yandira Mouriz Coca yandira@suss.co.cu

Tutor: Msc. Lydia Rosa Ríos Rodríguez

Sancti Spiritus, Junio del 2007
“Año 49 de la Revolución”

Resumen

El desarrollo que ha alcanzado la informática a nivel mundial en diferentes áreas, en particular en la educación y la experiencia acumulada en este campo, plantean la necesidad de investigar y profundizar en un conjunto de problemas inherentes a la informática educativa, que tenga la flexibilidad de ajustarse y modificarse según el avance de las tecnologías, el desarrollo de la sociedad cubana y el contexto de su aplicación.

El uso del software educativo se hace cada vez más evidente dentro del proceso de aprendizaje de los distintos niveles de enseñanza; el centro de estudio de este trabajo, específicamente es hacer una revisión del desarrollo de las Tecnologías de la información y la comunicación en la educación y proponer un simulador para apoyar el aprendizaje del proceso de matching en la asignatura programación lógica de la carrera Licenciatura en Ciencias de la Computación.

Se analizan las posibilidades que brindan el lenguaje Java y los applets para el desarrollo del simulador y se expone el diseño del mismo a través de la metodología RUP.

Palabras Claves: PROLOG, MATCHING, SIMULADOR

Índice

| | |
|--|-----------|
| INTRODUCCIÓN | 1 |
| CAPÍTULO 1 FUNDAMENTOS PARA EL USO DE UN SIMULADOR QUE POSIBILITE EL APRENDIZAJE DEL MATCHING | 10 |
| 1.1 TECNOLOGÍAS DE LA INFORMACIÓN Y LA COMUNICACIÓN | 10 |
| 1.2 TECNOLOGÍAS DE LA INFORMACIÓN Y LA COMUNICACIÓN EN LA EDUCACIÓN. | 12 |
| 1.3 CUBA Y LAS TECNOLOGÍAS DE LA INFORMACIÓN Y LA COMUNICACIÓN. | 14 |
| 1.4 SOFTWARE EDUCATIVO. SIMULADORES | 16 |
| 1.5 APRENDIZAJE DE PROLOG. MATCHING. | 18 |
| 1.6 ROL DEL PROFESOR FRENTE AL ROL DEL ESTUDIANTE. | 21 |
| 1.7 FUNDAMENTACIÓN DE LA PROPUESTA DE DISEÑO DEL SIMULADOR | 23 |
| 1.8 METODOLOGÍA PARA EL DISEÑO DEL SIMULADOR | 26 |
| 1.9 TECNOLOGÍAS PARA EL DESARROLLO DEL SIMULADOR | 30 |
| 1.9.2 MAPAS CONCEPTUALES | 30 |
| 1.9.3 WORLD WIDE WEB | 30 |
| 1.9.4 JAVA. APPLET | 31 |
| 1.9.5 EASY JAVA SIMULATIONS | 33 |
| 1.9.6 ECLIPSE | 34 |
| 1.9.7 NETBEANS | 35 |
| 1.9.8 MACROMEDIA FLASH | 37 |
| 1.9.9 W-PROLOG | 38 |
| CAPÍTULO 2 ANÁLISIS Y DISEÑO DE LA PROPUESTA DE SOLUCIÓN | 39 |
| 2.1 CARACTERIZACIÓN SOBRE ASPECTOS GENERALES DE LA ESTRUCTURA, ORGANIZACIÓN Y FUNCIONAMIENTO DEL CONTEXTO DONDE SE DESARROLLÓ EL ESTUDIO. | 39 |
| 2.2 DIAGNÓSTICO INICIAL DEL PROBLEMA PARA DESARROLLAR LA PROPUESTA DE DISEÑO DEL SIMULADOR QUE MEJORE LA COMPRENSIÓN DEL PROCESO DE MATCHING. | 40 |
| 2.3 DESCRIPCIÓN DEL NEGOCIO. | 45 |
| 2.3.1 REPRESENTACIÓN DE LOS CASOS DE USO DEL NEGOCIO. | 45 |
| 2.3.4 DIAGRAMA DE ACTIVIDADES. | 48 |
| 2.3.5 DIAGRAMA DE CLASES DEL MODELO DE OBJETOS. | 49 |
| 2.4 ESPECIFICACIÓN DE LOS REQUISITOS DE SOFTWARE. | 50 |
| 2.5 DESCRIPCIÓN DEL SISTEMA PROPUESTO. | 52 |
| 2.5.2 DIAGRAMA DE CASOS DE USOS DEL SISTEMA. | 53 |
| 2.5.4 DIAGRAMA DE CLASES. | 54 |
| 2.5.5 DIAGRAMAS DE INTERACCIÓN. | 59 |
| 2.6 DIAGRAMA DE NAVEGACIÓN | 60 |
| 2.7 ALGUNOS CRITERIOS DE CALIDAD A TENER EN CUENTA PARA DISEÑAR EL SIMULADOR | 61 |
| 2.8 VALORACIÓN DE LA FACTIBILIDAD DEL DISEÑO DEL SIMULADOR POR EL CRITERIO DE EXPERTOS. | 62 |
| CONCLUSIONES | 65 |
| RECOMENDACIONES | 66 |
| BIBLIOGRAFÍA | 67 |
| ANEXOS | 73 |

Introducción

Desde tiempos remotos el hombre ha buscado diferentes métodos, herramientas, técnicas para facilitar la realización de sus actividades; esto no solo le ha permitido conocer el mundo sino que lo ha ido transformando. Dentro de la esfera de la educación, específicamente en el proceso de enseñanza-aprendizaje de las diferentes asignaturas, también ocurre lo mismo; en este caso, existen disímiles medios de enseñanza para facilitar cada vez más este proceso, entre los que se destacan las láminas, maquetas, libros, transparencias, retroproyectores, la radio, la televisión y últimamente las computadoras.

En la actualidad los docentes están preocupados por dotar a los alumnos de herramientas y recursos de aprendizaje que faciliten su autonomía personal, y que los preparen como profesionales competentes y flexibles, capaces de enfrentarse a los constantes cambios científicos y tecnológicos que está viviendo la humanidad.

La aplicación de las Tecnologías de la Información y la Comunicación (TIC) en el proceso de enseñanza aprendizaje actual y las diferentes formas en que se debe abordar este problema son tema de interés investigativo a la hora de perfeccionar la formación del profesional. Para formar un profesional cuyo aval curricular esté acorde con las exigencias modernas mundiales y no al margen de estas, se debe lograr elevar, cada vez más, la efectividad del proceso docente y ello en gran medida se logra con la incorporación de TIC en el desarrollo de los diferentes cursos.

El uso de la computadora como herramienta del proceso de enseñanza - aprendizaje tiene varias aristas que parten del doble carácter que posee la computadora como medio de enseñanza y como herramienta de trabajo.

La enseñanza y el aprendizaje a través de máquinas no es algo de reciente invento. Al principio se comenzó por el radio a quien le sucedieron la televisión, el cine, el video y

la computadora desde los más simples software hasta las más recientes aplicaciones de multimedia e INTERNET. ¹

La docencia en todo el mundo ha experimentado el uso de todas estas tecnologías con el fin de amenizar el proceso de aprendizaje y mejorar la calidad de la enseñanza, procurando siempre sacar provecho de la combinación estratégica hombre - máquina.

La docencia asistida por computadoras como todo recurso para ayudar a aprender tiene sus lados positivos y negativos. La esencia positiva radica en que permite la simulación de procesos en tiempo real de forma interactiva, desarrolla el conocimiento lógico y lo organiza, además permite economizar gran cantidad de tiempo con respecto al uso de otros medios tradicionales como la retrotransparencia o las diapositivas e incorporan efectos especiales de vídeo y sonido que amenizan el aprendizaje. ²

La incorporación de las TIC permite un estudio individual más eficiente y así como la evaluación y a la par mejora la calidad de los trabajos presentados por los estudiantes y posibilita también disminuir el costo de la enseñanza al introducirse una gran cantidad de materiales de consulta como folletos, libros, guías de autopreparación, entre otros. en soporte magnético de donde se imprime solamente aquella información que le sea necesaria.

Lo negativo está definido por el factor humano: la efectividad del aprendizaje mediante máquinas será alta o baja en dependencia de las posibilidades comunicativas y los conocimientos de las personas que se valgan de ellas para enseñar o para aprender.

En el mundo actual la escuela, sobre todo los centros de educación superior, tiende cada vez más a ser un centro de gestión y gerencia del conocimiento que incorpora de manera vertiginosa los recursos de las TIC.

La utilización de las posibilidades que ofrecen estas nuevas tecnologías propician el establecimiento de formas de comunicación alternativas alumno - profesor. Ejemplo de

¹ Escalona, Miguel. Disponible en <http://www.campus-oei.org/revista/deloslectores/997Escalona.PDF>

² Almeida S., Febles J., Bolaños O. Disponible en http://www.bvs.sld.cu/revistas/ems/vol11_1_97/ems05197.htm

ello lo constituye la utilización del correo electrónico como medio de consulta y asesoría o como forma de recepción de tareas extraclases. Otros recursos como INTERNET han presentado nuevas alternativas a la docencia. Surgen modalidades nuevas de enseñanza como la educación virtual o educación a distancia.

La enseñanza a distancia ha proliferado en todo el mundo como una consecuencia inmediata del desarrollo de las TIC y como una respuesta al incremento de la población a nivel mundial y las dificultades para que los educandos puedan acceder a los centros educacionales o más bien para que los centros educacionales se acerquen, cada vez más, a los lugares donde residen las personas que deben recibir sus beneficios.³

Las experiencias emanadas de la educación a distancia asistida y la aplicación en mayor escala de las TIC deberán revertirse en un futuro inmediato en los cursos de la educación a distancia con vistas a mejorar cualitativa y cuantitativamente los resultados de este tipo de enseñanza.

Los docentes que no sean capaces de entender la diferencia entre el aprender de y con las nuevas tecnologías no podrán introducir cualitativamente, independientemente de la preparación y la vocación que tenga para ello, las TIC como instrumento y su gestión favorecerá tan solo su introducción como materia en el desarrollo de las asignaturas que imparten.

Las Tecnologías de la Información y la Comunicación deben utilizarse para la elevación de la calidad de la docencia pues motivan y hacen el aprendizaje más ameno, lo cual logra elevar la eficiencia del proceso docente - educativo e, indiscutiblemente, elevan a la par la calidad de la asimilación de los contenidos. La enseñanza presencial y la virtual podrán, definitivamente, coexistir en el siglo XXI pues cada una tiene sus funciones específicas y desempeñan un determinado rol en la formación preprofesional o continúa.

³ González, Y. Disponible en <http://www.campus-oei.org/revista/rie24a04.htm>

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

En dependencia de estas características del software educativo se ha venido estableciendo una agrupación y una clasificación de los mismos tomando como elemento clasificador la función que realizan dentro del proceso docente. Es usual encontrar en la literatura clasificaciones como la siguiente: Tutoriales, Entrenadores, Repasadores, Evaluadores, Simuladores, Libros electrónicos, Juegos Instructivos.

Se debe 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.⁴

En Cuba se imparte la asignatura de Programación Lógica (PL) en la carrera Licenciatura en Ciencias de la Computación, la cual es fundamental para el desarrollo de la inteligencia artificial. Especialistas en el tema y profesores que imparten esta asignatura indican que a los estudiantes les resulta muy difícil comprender los contenidos de la misma. La enseñanza de la Programación Lógica enfrenta el reto de recibirse después que se conocen otros paradigmas de programación y donde los estudiantes están acostumbrados a pensar de una forma que se diferencia mucho del estilo y la manera de pensar que se exige en el nuevo paradigma.

Los temas que se abordan resultan muy importantes para el desarrollo de habilidades relacionadas con la programación declarativa, a la vez que son complejos y diversos. La calidad y la profundidad de los resultados docentes en esta asignatura se ven limitados, al igual que la comprensión de algunos conceptos y procesos de la misma de acuerdo con declaraciones del personal calificado. Uno de estos procesos es el de matching. El

matching de términos es la base de muchos trabajos en deducción automática y es el mecanismo mediante el cuál las variables lógicas toman valor en Prolog.

En esta dirección se ha realizado una búsqueda, con el objetivo de encontrar algún tipo de software educativo, que caracterizado por su novedad, le ofrezca al estudiante una mejor comprensión del proceso de matching y que esto a su vez constituya un apoyo para el proceso de enseñanza-aprendizaje del Prolog. Hasta el momento no hay evidencias de que se haya realizado un software con estas características, o al menos no esta a disposición en las universidades cubanas.

En consecuencia se determinó el siguiente **problema científico** ¿Cómo contribuir a mejorar el aprendizaje del proceso de matching de la programación lógica?

Para darle respuesta al problema planteado se toma como **objeto de estudio** el proceso aprendizaje de la programación lógica en la carrera de Licenciatura en Ciencias de la Computación y como **campo de acción** el aprendizaje del proceso de matching de la Programación Lógica.

Esta investigación tiene como **objetivo general** diseñar un simulador que contribuya a mejorar el aprendizaje del proceso de matching de la Programación Lógica en estudiantes de la carrera de Licenciatura en Ciencias de la Computación.

Para la búsqueda del conocimiento científico se formularon las siguientes **Preguntas Científicas**

1. ¿Cuáles son los fundamentos teóricos y metodológicos que sustentan el uso de un medio informático en aprendizaje del proceso de matching?
2. ¿Cómo se desarrolla el aprendizaje del proceso de matching en la asignatura Programación Lógica?
3. ¿Cómo diseñar un simulador para mejorar el aprendizaje del proceso de matching en la asignatura Programación Lógica?

⁴ Rivero, Alfonso. Disponible en

4. ¿Cómo se puede validar la efectividad del simulador?

Para responder las preguntas científicas se plantean las **Tareas de Investigación**

1. Sistematización de los fundamentos teóricos y metodológicos relacionados con los medios informáticos para mejorar el aprendizaje del proceso de matching de la Programación Lógica.
2. Diagnóstico sobre el estado actual del aprendizaje del proceso de matching de la Programación Lógica.
3. Diseño de un simulador para mejorar el aprendizaje del proceso de matching en la asignatura Programación Lógica.
4. Validación del simulador por criterio de expertos.

En la concepción de esta investigación se utilizaron un sistema de métodos y técnicas de la investigación educacional con sus respectivos instrumentos. Para la construcción y desarrollo de la fundamentación teórica se procedió a través del nivel teórico y se utilizaron los siguientes métodos:

Histórico-Lógico: La investigación parte de un análisis histórico evolutivo del desarrollo del proceso de aprendizaje de la asignatura Programación Lógica, el método sirvió para analizar a través de diferentes fuentes de información la evolución y marcha progresiva de las teorías fundamentales sobre las cuales se realiza la investigación.

Análisis y Síntesis: Se realizó un análisis de diferentes ideas del estudio realizado y sintetizar los elementos que resultaron útiles.

Inducción y Deducción: Este método se aplicó para estudiar el modelo actuante, precisando de su comportamiento de lo particular a lo general; así como el

razonamiento de lo general a lo particular. La deducción permitió determinar el problema científico partiendo de los criterios de los diferentes autores y de la teoría científica.

De los métodos del nivel empírico se emplearon los siguientes

La Observación: Permitió conocer el problema y el objeto de investigación, para obtener información y sistematizar aspectos relacionados con el proceso de aprendizaje de la asignatura Programación Lógica.

A través de la encuesta se pudo conocer el estado y características del proceso de aprendizaje de la asignatura Programación Lógica.

La entrevista: Se utilizó como técnica de interrogación, la entrevista no estandarizada, donde los entrevistados narraron sus puntos de vista, criterios, y vivencias.

el criterio de especialistas se hizo necesario para estimar la validez, pertinencia y posible efectividad de la propuesta realizada.

El análisis documental se utilizó para explorar en las fuentes del conocimiento la información que existe sobre el objeto de estudio y permitió realizar un análisis en la documentación que recoge las metodologías y aspectos a tener en cuenta para la realización de la propuesta.

Los métodos de nivel matemático y estadístico permitieron la cuantificación y el procesamiento de los datos para su interpretación.

Cálculo porcentual: Se utilizó para procesar cuantitativamente la información y medir la confiabilidad y validez de los instrumentos aplicados. De la estadística descriptiva se utilizaron los gráficos para ilustrar los datos obtenidos.

Todos los estudiantes de la carrera de Licenciatura en Ciencias de la Computación, de la facultad de Matemática, Física y Computación, de la Universidad Central "Martha

Abreu" de las Villas que reciben la asignatura de Programación Lógica constituyen la población. Debido a que esta investigación forma parte de un proyecto que se está desarrollando en esta universidad y es donde estará ubicado finalmente el simulador.

La selección de la muestra es de tipo intencional y corresponde a los 36 estudiantes de la población lo que constituye el 100% por lo que garantiza la representatividad de la población.

La **actualidad del tema** se basa en la importancia que tiene la PL en el desarrollo de la inteligencia artificial, así como la gestión del conocimiento por parte de los alumnos. En el presente trabajo se propone un simulador para comprender el proceso de matching que es uno de los fundamentales en la solución de problemas de la PL.

La **novedad científica** de este trabajo radica en que aporta un software educativo que contribuye a dar cumplimiento al objetivo del proceso de enseñanza y aprendizaje de la PL. Con lo cual el simulador facilita el aprendizaje del proceso de matching que es la base de la PL y no existe otro antecedente a este que ha sido objeto de diseño y estudio.

La **contribución práctica** de la tesis es el diseño e implementación de un simulador que será usado para mejorar el desarrollo de la asignatura Programación Lógica (PROLOG) y más específicamente para una mejor comprensión del proceso de matching. El mismo promueve el desarrollo de destrezas en la búsqueda y manipulación de información, y el desarrollo de habilidades para la toma de decisiones, procesos de pensamiento, de apropiación del conocimiento. Desarrolla componentes de la memoria audiovisual de los estudiantes. Permite que el estudiante tenga un papel preponderante ya que se autoinstruye, autoprograma, recibe un aprendizaje individualizado de acuerdo con su ritmo individual de asimilación.

El trabajo posee una introducción, dos capítulos, conclusiones, recomendaciones, bibliografía y anexos. La introducción expresa los aspectos más generales de la

investigación del diseño teórico y metodológico.

El capítulo 1 aborda el marco teórico de esta investigación y se hace un análisis de las TICs en la educación. En el capítulo 2 se describen las principales herramientas y la metodología a utilizar, se presenta el diseño de la propuesta de solución, se exponen los requisitos del sistema y la validación del criterio de especialistas.

Capítulo 1 Fundamentos para el uso de un simulador que posibilite el aprendizaje del matching

Introducción

Las TIC se han transformado en una nueva manifestación cultural además de mantener su valor inicial de medio de conservación y transmisión de la información. Su influencia se refleja en todas las esferas de la vida, en particular su introducción en la enseñanza ha obligado a repensar en los métodos y los medios de enseñanza.

Cuba, enfrascado en la noble tarea de lograr una sociedad más culta y preparada para los retos del futuro ha dado y está dando pasos cruciales que llevan a extender la enseñanza, a cualquier nivel escolar, hasta lugares impensados en la mayoría del mundo.

En este capítulo se sistematizan las principales características, definiciones y tendencias que establecerán el desarrollo del producto educativo objeto de estudio.

1.1 Tecnologías de la Información y la Comunicación

Resulta innegable el auge cada vez mayor de las TIC en las diferentes esferas de la sociedad a escala mundial. El desarrollo impetuoso de la ciencia y la tecnología ha llevado a la sociedad a entrar al nuevo milenio inmerso en lo que se ha dado en llamar “era de la información”. Sin lugar a dudas, se está en presencia de una revolución tecnológica de alcance insospechado.

Pero ¿Qué son las TIC? Existen muchas definiciones al respecto, una de ellas es la que la define como “...Un conjunto de aparatos, redes y servicios que se integran, en un sistema de información interconectado y complementario. La innovación tecnológica consiste en que se pierden la frontera entre un medio de información y otro”⁵. Estas TIC conforman un sistema integrado por:

⁵ Gómez, C.1992

Las telecomunicaciones: Representadas por los satélites destinados a la transmisión de señales telefónicas, telegráficas y televisivas; la telefonía que ha tenido un desarrollo impresionante a partir del surgimiento de la señal digital; el fax y el modem; y por la fibra óptica, nuevo conductor de la información en forma luminosa que entre sus múltiples ventajas económicas se distingue el transmitir la señal a grandes distancias sin necesidad de usar repetidores y tener ancho de banda muy amplio.

La informática: Caracterizada por notables avances en materia de hardware y software que permiten producir, transmitir, manipular y almacenar la información con mas efectividad, distinguiéndose la multimedia, las redes locales y globales (INTERNET), los bancos interactivo de información, los servicios de mensajería electrónica, etc.

La tecnología audiovisual: Que ha perfeccionado la televisión de libre señal, la televisión por cable, la televisión restringida (pago por evento) y la televisión de alta definición.

Las TIC designan a la vez un conjunto de innovaciones tecnológicas pero también las herramientas que permiten una redefinición radical del funcionamiento de la sociedad.

Las TIC son aquellas herramientas computacionales e informáticas que procesan, almacenan, sintetizan, recuperan y presentan información representada de la más variada forma.⁶

Las definiciones que se han ofrecido de las nuevas tecnologías de la información y comunicación, son muy variadas y en todas ellas se les tiende a considerar como aquellos instrumentos técnicos que giran en torno a la información y a los nuevos descubrimientos que sobre las mismas se vayan originando.

La amplia utilización de las TIC en el mundo, ha traído como consecuencia un importante cambio en la economía mundial, particularmente en los países más industrializados, sumándole a los factores tradicionales de producción para la

generación de riquezas, un nuevo factor que resulta estratégico: El conocimiento. Es por eso que ya no se habla de la “sociedad de la información”, sino también de la “sociedad del conocimiento”. Sus efectos y alcance sobrepasan los propios marcos de la información y la comunicación, y puede traer aparejadas modificaciones en las estructuras políticas, social, económica, laboral y jurídica debido a que posibilitan obtener, almacenar, procesar, manipular y distribuir con rapidez la información.

1.2 Tecnologías de la información y la comunicación en la educación.

Esta revolución, encabezada con la introducción de las tecnologías de la información y la comunicación en casi todos los ámbitos de la vida diaria, afecta a todos de una forma u otra. Y es que, aparte de las propias actitudes ante ellas, es indudable, y casi inevitable, el impacto que sobre los procesos informativos y comunicativos está teniendo las denominadas tecnologías, con un alcance generalizado sobre todas las actividades y ámbitos del ser humano, desde esferas macro y micro económicas, políticas, sociales, culturales, laborales, o formativas, hasta incluso espacios más personales.

Una de las influencias más significativas de las TIC en las organizaciones educativas se va a producir como consecuencia de las posibilidades que éstas tienen para superar y romper las variables espacio-temporales en las cuales tiende a desenvolverse la formación actual, tanto la presencial como la a distancia, en lo que se puede considerar como circuitos tradicionales de enseñanza.⁷

La enseñanza tradicional se ha venido desarrollando con una coincidencia en el espacio y el tiempo entre la persona que enseña y aquella que aprende, llevándose a cabo normalmente en un lugar que viene a considerarse como aula, en la cual se realizan todas los intercambios entre aquellos que enseñan y los que aprenden, y todas las operaciones necesarias que indican que el alumno ha superado un nivel concreto y puede pasar a otra aula donde recibirá unos conocimientos formativos más profundos.

⁶ Arias, Leandro. Disponible en <http://www.monografias.com/trabajos13/simucom/simucom.shtml>

⁷ Herrera, René. Disponible en <http://www.somece.org.mx/memorias/2000/docs/453.DOC>

Frente a este tipo de propuesta la enseñanza a distancia vino a introducir la posibilidad de una ruptura entre ambas dimensiones, de manera que las personas que enseñaban y aprendía podían encontrarse en tiempos diferentes y lugares distintos.

Las tecnologías de la información y la comunicación permiten no sólo la disociación de dichas variables, sino también la posibilidad de la interacción entre los participantes en el acto comunicativo de la enseñanza, e interacción tanto sincrónica como asincrónica, de manera que el aprendizaje se producirá en un no lugar, como es el ciberespacio; es decir, “un espacio físico pero no real, en el cual se tienden a desarrollar interacciones comunicativas mediáticas.”⁸

De tal forma que emisores y receptores, establecerán en todas, pero en ninguna parte, espacios de encuentros para la comunicación superando las limitaciones espacio-temporales que la realidad física impone. Ello implicará que se puede interaccionar con otras personas ubicadas dentro de la red global de comunicaciones independientemente del lugar donde se ubique, facilitándose de esta forma el acercamiento entre las personas.⁹

Hoy en día los maestros que deciden emplear la computación como medio de enseñanza tienen a su disposición una amplia gama de programas que pueden ser empleados por ellos con ese propósito.

Unos han sido desarrollados expresamente con ese fin por equipos multidisciplinarios integrados por pedagogos, psicólogos, artistas y programadores, otros por solitarios programadores que se apoyan en sus conocimientos sobre su especialidad para apuntalar su discutible experiencia (o a veces intuición) pedagógica, y otros, son simples programas comerciales que por algunas de sus características pueden ser empleados con provecho dentro de la actividad docente.¹⁰

⁸ Kofman, Hugo. Disponible en <http://www.rieoei.org/deloslectores/430Kofman.pdf>

⁹ Cabrera, Julio. Disponible en <http://www.uib.es/depart/gte/edutec-e/revelec3/revelec3.html>

¹⁰ Programación lógica. Disponible en http://es.wikipedia.org/wiki/Programaci%C3%B3n_L%C3%B3gica

Cada uno de estos programas tiene propósitos específicos, dirigidos a contribuir con el desarrollo de alguno de los aspectos del proceso docente. Unos pretenden enseñar al alumno un contenido nuevo, otros simulan el desarrollo de un proceso físico, los hay que intentan contribuir al desarrollo de alguna habilidad, intelectual o motora; otros sólo pretenden evaluar los conocimientos del estudiante sobre un determinado contenido.

Con la computadora se le brinda al maestro la posibilidad de llevar a su clase la dinámica de la contemporaneidad, pues puede enriquecer mucho más la enseñanza de las asignaturas que imparte, ya que por esta vía tiene acceso a videos, diccionarios, imágenes, entre otras oportunidades que brinda el uso de la Informática.

Estos son los programas de computación que tienen como fin apoyar el proceso de enseñanza de aprendizaje contribuyendo a elevar su calidad y a una mejor atención al tratamiento de las diferencias individuales, sobre la base de una adecuada proyección de estrategia a seguir tanto en el proceso de implementación como en su explotación.

Entre los tipos de software educativo que existen se encuentran los tutoriales, entrenadores, repasadores, test o evaluadores, libros electrónicos, juegos instructivos, y los simuladores del cual se abundará en el presente trabajo.

El software educativo supera las barreras de la distancia: elimina la dificultad de traslado por falta de tiempo o condiciones físicas. Supera la barrera del tiempo: el estudiante no tiene que abandonar su hogar ni su lugar de empleo, puede estudiar en horario libre.

1.3 Cuba y las tecnologías de la información y la comunicación.

El desarrollo creciente de las tecnologías relacionadas con los sistemas computarizados tiene en la actualidad un impacto muy profundo en los procesos de enseñanza – aprendizaje de las universidades e Instituciones de Educación Superior,

presentándose estas tecnologías como una alternativa para satisfacer demandas de la educación a distancia, del aprendizaje permanente, de la diferenciación a través de la atención individualizada de los estudiantes en el proceso de asimilación u otros aspectos relacionados con este proceso que no podían ni soñarse hace algunos años. Al respecto se señala, “las Tecnologías de la Información y las Comunicaciones (TIC) están produciendo una revolución en las formas de producción y circulación del conocimiento, cambios en los modelos de pensamiento, en la estructura del conocimiento, en los procesos de enseñanza - aprendizaje...”¹¹

Para la Educación Superior cubana las TIC y su incidencia en el desarrollo social quedó claramente expresado por el Dr. Ing. Fernando Vecino Alegret cuando refiriéndose a las proyecciones de trabajo más importantes presentes y futuras de este Ministerio señaló:

“Otro aspecto significativo ha sido todo el programa actual y prospectivo en relación con la computación y las nuevas tecnologías de información, por las implicaciones que ello tiene sobre nuestra organización y que consiste en el dominio, uso y desarrollo de las técnicas computacionales, que van más allá de la potenciación de las capacidades de procesamiento y recuperación de información de todo tipo. Sin una estrategia en este sentido sería imposible lograr que en nuestras instituciones se formen profesionales competitivos altamente eficientes, ni desarrollar el postgrado con los requerimientos del nuevo siglo, ni tampoco alcanzar en la actividad científica mayor calidad y productividad.”¹²

Las Universidades cubanas han sido líderes históricos en la introducción y desarrollo en el país de las tecnologías computacionales. Este liderazgo se ha manifestado en la cantidad de resultados investigativos y productivo obtenidos en disciplinas relacionadas con la TIC y en los programas de formación de profesionales en los campos de la informática, electrónica y telecomunicaciones.

En Cuba se trabaja y se avanza en la incorporación de las TIC en un proceso de estudio y asimilación permanente de los modelos pedagógicos y tecnológicos que mejor

¹¹ Casas Armengol, Miguel. 1991

¹² Vecino Alegret, Fernando. 1996.

respondan a los presupuestos teóricos que sustentan el proyecto social educativo cubano.

1.4 Software Educativo. Simuladores

La literatura define el concepto genérico de Software Educativo como 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 una computadora en los procesos de enseñar y aprender.¹³

Los software educativo son aquellos programas capaces de servir de ayuda al aprendizaje del alumno y de apoyo, nunca de sustituto, a la labor pedagógica del profesor, y además, dadas las cualidades de los mismos (interacción, dinamismo, colorido, multimedia, etc.), posibilitadores de mejoras del aprendizaje del alumno.¹⁴

Finalmente según las definiciones anteriores, los Software Educativos se pueden considerar como el conjunto de recursos informáticos diseñados con la intención de ser utilizados en el contexto del proceso de enseñanza – aprendizaje.

Se caracterizan por ser altamente interactivos, a partir del empleo de recursos multimedia, como videos, sonidos, fotografías, diccionarios especializados, Explicaciones de experimentados profesores, ejercicios y juegos instructivos que apoyan las funciones de evaluación y diagnóstico.

El presente trabajo se centra en los simuladores: programas que intentan dar una representación lo más realista posible de un dominio complejo. Este tipo de software educativo intenta explicar un fenómeno mediante una representación controlada del mismo en la computadora con apariencia lo más cercana posible al fenómeno real.

¹³ Sánchez, Jaime. (1999).

¹⁴ Bezanilla Albisua, M^a J. y Martínez Ruiz, J. A. (1996).

El simulador es un programa que, simulando un modelo dado, le permite al usuario verlo funcionar y desplegar resultados. El modelo puede ser: un sistema de ecuaciones, un conjunto de procedimientos o una serie de reglas de acción condicionadas, como por ejemplo, para operar una planta de energía nuclear, poblaciones de plantas; procedimientos para guiar un cohete, etc.¹⁵

La simulación es el primer medio considerado interactivo, en el sentido de que otorga una retroalimentación intrínseca sobre las acciones del estudiante. De modo que la simulación le permite al estudiante tener una experiencia particular, que controla él mismo dado que al seleccionar los parámetros a cambiar, efectivamente determina el enfoque del tema.¹⁶

En los 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.¹⁷

Se puede concluir de todas las definiciones anteriores que los simuladores son software que se proponen enseñar sobre algún aspecto del mundo imitándolo. Un software de este tipo constituye un poderoso medio ya que el estudiante puede interactuar con el programa simulando situaciones reales. Los estudiantes no sólo se motivan por la simulación, sino que también aprenden interactuando con ella de manera similar a como pudieran hacerlo en situaciones reales.

El simulador será usado para mejorar el desarrollo de la asignatura Programación Lógica (PROLOG) y más específicamente para mejorar el aprendizaje del proceso de matching.

¹⁵ Arias, Leandro. Disponible en <http://www.monografias.com/trabajos13/simucos/simucos.shtml>

¹⁶ Spinak, Silvia; Maggioli, Susana. Disponible en www.crnti.edu.uy/05trabajos/normas1/software.doc

¹⁷ Alfonso, Ileana; Corral, Roberto.

Disponible cepes.uh.cu/bibliomaestria/tendenciaspedagogicas/libros/TENDE2.doc

El mismo promueve el desarrollo de destrezas en la búsqueda y manipulación de información, y el desarrollo de habilidades para la toma de decisiones, procesos de pensamiento, de apropiación del conocimiento.

Desarrolla componentes de la memoria audiovisual de los estudiantes. La velocidad de la percepción a través de la memoria audiovisual tiende a ser más elevada que la memoria motora por lo que las personas que se preparan a vivir en las futuras generaciones requieren desarrollar intensamente estas formas de la memoria para no encontrarse en desventajas con otros conciudadanos del mundo.

Permite que el estudiante tenga un papel preponderante ya que se autoinstruye, autoprograma, recibe un aprendizaje individualizado de acuerdo con su ritmo individual de asimilación.

Esto último no quiere decir que no se logre el desarrollo de habilidades comunicativas en el alumno, todo lo contrario, el obtiene retroalimentación del propio programa y del profesor, también puede compartir sus reflexiones, análisis, errores y aciertos con sus compañeros de curso.

La simulación puede jugar un papel relevante con vista a desarrollar las habilidades de los estudiantes y es de hecho la forma más popular de aprendizaje basado en computador en la educación universitaria.

Una de las razones principales para introducir las simulaciones en la instrucción, la posibilidad de mostrar las características dinámicas y la evolución temporal de los sistemas. En el caso del simulador a desarrollar, simula el proceso de matching explicado con anterioridad que ayudan a la comprensión de esta asignatura, ya que son procesos muy abstractos que regularmente el estudiante no entiende.

1.5 Aprendizaje de PROLOG. Matching.

El desarrollo de Prolog se inició en 1970 con Alain Coulmeauer y Philippe Roussel, quienes estaban interesados en desarrollar un lenguaje para hacer

deducciones a partir de texto. PROLOG es un lenguaje de programación para ordenadores se utiliza para resolver problemas en los que entran en juego objetos y relaciones entre ellos.¹⁸

Históricamente, los ordenadores se han programado utilizando lenguajes muy cercanos a las peculiaridades de la propia máquina: operaciones aritméticas simples, instrucciones de acceso a memoria, etc. Un programa escrito de esta manera puede ocultar totalmente su propósito a la comprensión de un ser humano, incluso uno entrenado.

Hoy día, estos lenguajes pertenecientes al paradigma de la Programación imperativa han evolucionado de manera que ya no son tan crípticos. Sin embargo, aún existen casos donde el uso de lenguajes imperativos es inviable debido a la complejidad del problema a resolver.

En cambio, la lógica matemática es la manera más sencilla, para el intelecto humano, de expresar formalmente problemas complejos y de resolverlos mediante la aplicación de reglas, hipótesis y teoremas. De ahí que el concepto de “programación lógica” resulte atractivo en diversos campos donde la programación tradicional es un fracaso.¹⁹

Prolog tiene una sintaxis y semántica simples. Puesto que busca relaciones entre una serie de objetos, la variable y la lista son las estructuras de datos básicas que se usan.²⁰

La Programación Lógica analiza las frases sencillas del lenguaje (fórmulas atómicas o elementales) separándolas en Términos y Predicados. Los términos hacen referencia a los objetos que intervienen y los predicados a las propiedades o relaciones entre estos objetos.

¹⁸ El lenguaje Prolog. <http://www.etse.urv.es/EngInf/assig/iai/Laboratoris/Prolog/IntroduccionProlog.pdf>

¹⁹ Programación lógica. Disponible en http://es.wikipedia.org/wiki/Programaci%C3%B3n_L%C3%B3gica

²⁰ El lenguaje Prolog. <http://www.etse.urv.es/EngInf/assig/iai/Laboratoris/Prolog/IntroduccionProlog.pdf>

Los términos pueden ser constantes o variables, y suponemos definido un dominio no vacío en el cual toman valores (Universo del Discurso). Para saber cuántos individuos del universo cumplen una determinada propiedad o relación, cuantificamos los términos.

Las constantes se utilizan para dar nombre a objetos concretos del dominio, dicho de otra manera, representan individuos conocidos de nuestro Universo. Además, como ya hemos dicho, las constantes atómicas de Prolog también se utilizan para representar propiedades y relaciones entre los objetos del dominio.

Las variables se utilizan para representar objetos cualesquiera del Universo u objetos desconocidos en ese momento, es decir, son las incógnitas del problema. Se diferencian de los átomos en que empiezan siempre con una letra mayúscula o con el signo de subrayado (_). Así, deberemos ir con cuidado ya que cualquier identificador que empiece por mayúscula, será tomado por Prolog como una variable.²¹

El mecanismo de operación más importante que se realiza sobre términos es el matching. El matching es la unificación de los términos en Prolog. Dos términos pueden hacer matching cuando:

- 1- Un átomo solo puede unificarse con otro átomo exactamente igual a él.
- 2- Una variable libre se instancia con cualquier termino.
- 3- Una estructura se instancia solo con otra estructura que tenga el mismo nombre (denominado functor) y la misma cantidad de argumentos (a esa cantidad se le denomina el arity de la estructura).²²
 - i) El matching de términos es la base de muchos trabajos en deducción automática y el uso de la inferencia lógica en Inteligencia Artificial y es el mecanismo mediante el cuál las variables lógicas toman valor en Prolog.

²¹ Llorens Largo, Faraón; Castel de Haro, M^a Jesús. Disponible en <http://www.dccia.ua.es/logica/prolog/docs/prolog.pdf>

Constituye uno de los mecanismos esenciales de Prolog, y consiste en buscar instancias comunes a dos átomos, uno de los cuales está en la cabeza de una cláusula y el otro en el cuerpo de otra cláusula.²³

En Lógica de Predicados se necesita un algoritmo para que compare dos términos y descubra si existe un conjunto de sustitución que los haga idénticos. Esta es la función de dicho algoritmo. Un unificador de términos es una sustitución que hace a los términos idénticos.²⁴

Aprender el lenguaje PROLOG es un gran reto debido a que el paradigma de programación difiere, en gran medida, de los paradigmas anteriormente aprendidos por los estudiantes, los cuales intentan extrapolar las ideas de los paradigmas conocidos.

Prolog constituye un nuevo estilo de programación al que el estudiante hasta este momento no se había enfrentado. Esta asignatura constituye una de las más difíciles a la que se enfrentan los estudiantes de la carrera, precisamente por lo abstracta que resulta su explicación y entendimiento.

Muchos de los conceptos que se manejan no son visibles a los estudiantes pues los realiza la computadora internamente, lo que hace que varios procesos no resulten del todo entendible al alumno, dificultando el exitoso desarrollo de la asignatura.

1.6 Rol del profesor frente al rol del estudiante

Esta incorporación de las nuevas tecnologías en los procesos de formación traerá consigo un cambio de los roles tradicionales desempeñado por los profesores.

La tarea pedagógica del profesor es compleja, en ella intervienen diversos elementos relacionados con su actividad y la del estudiante, donde están presentes los rasgos de la personalidad de ambos, así como las cualidades de la ciencia que enseña y los

²² García Valdivia, Zoila; Bello, Rafael.(2000)

²³ Gálvez, Daniel. Programación Lógica.

²⁴ Unificación. Disponible en http://www.monografias.com/trabajos/iartificial/pagina4_21.htm

aspectos relacionados con la forma de abordar didácticamente los programas de enseñanza.

El profesor a través de su actividad pedagógica facilita la construcción del conocimiento y el desarrollo de habilidades, así como, la formación de actitudes en las condiciones socio-históricas en las que se desenvuelve.

El profesor dejará a un lado su faceta de “trasmisor del conocimiento” para cumplir el papel de “conductor de estudiantes” enseñándoles como seleccionar los contenidos relevantes, asimilarlos, interrelacionarlos y ponerlos en práctica.

Asemejándose la calidad de la enseñanza, con la fidelidad con que el alumno repite la información presentada por el profesor. Por el contrario, propuestas como las aquí apuntadas tienden a favorecer la autoinstrucción por parte del estudiante y el trabajo cooperativo entre ellos, en la búsqueda constructivista, y no memorística, de los contenidos, habilidades y resolución de problemas.

En contraposición, a la ruptura del rol de transmisor de información, se tenderá a potenciar otros como el de evaluador, organizador de situaciones mediadas de aprendizaje, y diseñador de medios y materiales adaptados a las características de sus estudiantes y potencialidades de la tecnología que sea usada.

Lo cual exigirá nuevas respuestas organizativas, entre otros motivos, por que el profesor La formación y desarrollo de valores constituye un objetivo esencial en la educación cubana. A ella se dedica atención especial en los diferentes niveles de enseñanza a través de la concepción e instrumentación de proyectos educativos en los que el profesor ocupa un papel rector en el sistema de influencias que propician la formación y desarrollo de valores en los estudiantes.²⁵

²⁵ Torres, R.M. Disponible en http://www.unesco.cl/medios/biblioteca/documentos/analisis_prospectivas_lac_esp.pdf?menu=/esp/biblio/docdig

A través del simulador el profesor tiene función de facilitador del aprendizaje, propiciando las condiciones para la libre expresión de las potencialidades del estudiante.

El profesor diseña las situaciones de aprendizaje que estimulan la formación y desarrollo de valores como reguladores de la actuación del estudiante, en condiciones de interacción social. Ello implica diseñar tareas de aprendizaje que propicien la reflexión del estudiante en torno a la importancia de los valores que se pretenden formar para su vida personal y profesional, tareas que los estimulen a asumir una posición activa, flexible y perseverante en la expresión de sus valores en un clima de participación democrática sustentada en la aceptación, la confianza y el respeto a la diversidad de opiniones y en el que el profesor es una autoridad no impuesta sino ganada por su ejemplaridad y condición de modelo educativo.

Para el diseño de la herramienta en cuestión se ha realizado un trabajo importante de planificación y coordinación del equipo de profesores que están desarrollándola, debido a que cada uno de los ejercicios tiene su correspondiente nivel de complejidad, sensible a los avances progresivos del estudiante.

El profesor interviene e interactúa con los estudiantes durante el proceso de aprendizaje mediante el simulador, anticipando la resolución de posibles confusiones y asegurando una construcción adecuada en los momentos claves. Esta comunicación puede ser personal o utilizando las nuevas tecnologías de la información.

Con los simuladores el profesor genera artificialmente situaciones conflictivas que permitirán la formación de acciones de respuesta correcta que quedarán posteriormente registradas como experiencias en los centros emocionales y en la memoria más racional de los lóbulos prefrontales del individuo.

1.7 Fundamentación de la propuesta de diseño del Simulador

El hombre llega a conocer su entorno material y social, del cual es parte o elemento

inseparable, mediante un complejo proceso de aprendizaje el cual lo educa y capacita para interactuar con ellos de manera lógica y dialéctica.

La Pedagogía ha evolucionado mucho desde su origen etimológico que significaba conducir o llevar a un niño en el sentido espiritual o enseñarlo. Es una ciencia de carácter psicosocial que tiene por objeto el estudio de la educación con el fin de conocerlo y perfeccionarlo. También es una ciencia de carácter normativo porque no se dedica a describir el fenómeno educacional sino a establecer las pautas o normas que hemos de seguir para llevar a buen término dicho fenómeno.²⁶

Estas pautas o normas no son más que las tendencias pedagógicas. Con ellas se normalizan las formas del trabajo que se ha de desarrollar en el proceso de enseñanza facilitador de un aprendizaje eficiente, el papel que se le asigna tanto al educando como al educador.²⁷

Por eso debemos tener en cuenta la tendencia pedagógica que se asumirá a la hora de diseñar un software educativo. A continuación se hace un pequeño estudio de las mismas.

Enfoque histórico-cultural

Según Vigotski la esencia de la tendencia pedagógica del enfoque histórico-cultural es una concepción dirigida en lo fundamental a la enseñanza, facilitadora de un aprendizaje desarrollador, en dinámica interacción entre el sujeto cognoscente 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.²⁸

²⁶ Pedagogía. Disponible en <http://es.wikipedia.org/wiki/Pedagog%C3%ADa>

²⁷ Lino, Troadio; Alfonso, Ileana. Disponible en <http://www.monografias.com/trabajos6/tenpe/tenpe3.shtml>

²⁸ Lino, Troadio; Alfonso, Ileana. Disponible en <http://www.monografias.com/trabajos6/tenpe/tenpe3.shtml>

Precisamente el simulador será en este caso el mediador, o sea, la herramienta que el estudiante tomará de su entorno social, sobre la cual se apoyará para el estudio y le facilitará el aprendizaje.

Para 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, de allí la importancia que se adjudica a la actividad conjunta, a la relación de cooperación entre los alumnos y entre éstos y el profesor.

Esta concepción cambia la tradicional relación entre autoridad y distancia existente entre ambos participantes del proceso, señala como función fundamental del profesor la orientación y guía del estudiante, con el fin de potenciar sus posibilidades y convertir en realidad las potencialidades de su Zona de Desarrollo Próximo (ZDP).

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.²⁹

Tomando en consideración lo que plantea Vigotski en su teoría de la Zona de Desarrollo Próximo (ZDP), el simulador constituirá el nivel de ayuda que facilitará el camino para llegar a la zona de desarrollo próximo al minimizar el esfuerzo para el entendimiento.

En toda la bibliografía consultada se plantea que las simulaciones y los llamados micromundos, se relacionan con el paradigma constructivista pues como se planteo anteriormente el constructivismo sostiene que el alumno construye su conocimiento, a partir de su propia forma de ser, pensar e interpretar la información.

Pero como Vygotsky es considerado el precursor del constructivismo social, lo que en su enfoque considera al individuo como el resultado del proceso histórico y social donde el lenguaje desempeña un papel esencial. Él plantea que el conocimiento es un

proceso de interacción entre el sujeto y el medio, pero el medio entendido social y culturalmente, no solamente físico, como lo considera primordialmente Piaget.³⁰

Por estas razones para el diseño del simulador se asume un enfoque histórico cultural. El simulador facilita la enseñanza individualizada y el alumno no se inhibe de preguntar o de repetir varias veces la misma lección. Aumenta la motivación y el gusto por aprender. Promueve la enseñanza y aprendizaje activos, se estimula a buscar el conocimiento. También incrementa la retención por la combinación de imágenes, gráficos, etc., así como la presentación de situaciones de la vida real.

En el simulador el aprendizaje se concibe como el proceso de construcción, por parte del sujeto que aprende, de conocimientos, habilidades y motivos de actuación que se produce en condiciones de interacción social, en un medio sociohistórico concreto sobre la base de la experiencia individual y grupal y que lo conduce a su desarrollo personal.

Esta concepción de aprendizaje plantea ante todo el reconocimiento del carácter activo del estudiante en el proceso de construcción del conocimiento, su desarrollo en condiciones de interacción social, así como el hecho de que se aprenden no sólo conocimientos y habilidades, sino también valores y sentimientos que se expresan en la conducta del hombre como motivos de actuación.

1.8 Metodología para el diseño del Simulador

1.8.1 Desarrollo basado en RUP bajo la herramienta Rational Rose.

Cada día la producción de software busca adecuarse más a las necesidades del usuario, y como consecuencia aumenta el tamaño y complejidad de las aplicaciones. Por lo que para lograr la rentabilidad de estas producciones se necesita un proceso que integre las múltiples facetas de desarrollo del mismo.

²⁹ Percepción del aprendizaje desde el adulto mayor. Disponible en <http://www.monografias.com/trabajos29/aprendizaje-adulto-mayor/aprendizaje-adulto-mayor.shtml>

³⁰ Romo, Abel. El enfoque sociocultural del aprendizaje de Vygotsky. Disponible en <http://www.monografias.com/trabajos10/enso/enso.shtml>

El Proceso Unificado 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.³¹

En la metodología RUP los desarrolladores comienzan capturando los requisitos del cliente en la forma de casos de uso, creando el diagrama de casos de uso del negocio, diagrama de actividades, diagrama de clases del modelo de objetos. Después analizan y diseñan el sistema para cumplir los casos de uso, confeccionando el diagrama de casos de usos del sistema, diagrama de clases del sistema, diagramas de interacción, diagrama de navegación.

Después en la implementación se empieza con el resultado del diseño y se implementan las clases y subsistemas encontrados durante el diseño. Las clases se implementan como componentes de fichero que contienen código fuente.

Esta es la metodología que se sigue para el diseño e implementación del simulador.

UML (Unified Modeling Language).

UML (Unified Modeling Language) o Lenguaje de Modelación Unificado es un lenguaje gráfico para especificar, construir, visualizar y documentar las partes o artefactos (información que se utiliza o produce mediante un proceso de software). Pueden ser

³¹ Barreiro, Enrique. Disponible en <http://trevinca.ei.uvigo.es/~ebalonso/asignaturas/esx/guiones/esxClase4.pdf>

artefactos: un modelo, una descripción que comprende el desarrollo de software que se basen en el enfoque Orientado a Objetos, utilizándose también en el diseño Web. UML usa procesos de otras metodologías, aprovechando la experiencia de sus creadores, eliminó los componentes que resultaban de poca utilidad práctica y añadió nuevos elementos.³²

UML es un lenguaje más expresivo, claro y uniforme que los anteriores definidos para el diseño Orientado a Objetos, que no garantiza el éxito de los proyectos pero si mejora sustancialmente el desarrollo de los mismos, al permitir una nueva y fuerte integración entre las herramientas, los procesos y los dominios.

De forma general las principales características son:

- Lenguaje unificado para la modelación de sistemas
- Tecnología orientada a objetos
- El cliente participa en todas las etapas del proyecto
- Corrección de errores viables en todas las etapas
- Aplicable para tratar asuntos de escala inherentes a sistemas complejos de misión crítica, tiempo real y cliente/servidor

UML es desde finales de 1997, un lenguaje de modelado orientado a objetos estándar, de acuerdo con el Object Management Group, siendo utilizado diariamente por grandes organizaciones como: Microsoft, Oracle, Rational.

Rational Rose.

Existen herramientas Case de trabajo visuales como el Analise, el Designe y el Rational Rose, que permiten realizar el modelado del desarrollo de los proyectos, en la actualidad la mejor y más utilizada en el mercado mundial es Rational Rose y es la que se utiliza en la modelación de este proyecto.

³² Salinas Caro, Patricio. Disponible en <http://www.dcc.uchile.cl/~psalinas/uml/introduccion.html>

La Corporación Rational ofrece un Proceso Unificado (RUP) para el desarrollo de los proyectos de software, desde la etapa de Ingeniería de Requerimientos hasta la de pruebas. Para cada una de estas etapas existe una herramienta de ayuda en la administración de los proyectos, Rose es la herramienta del Rational para la etapa de análisis y diseño de sistemas.

Rose es una herramienta con plataforma independiente que ayuda a la comunicación entre los miembros de equipo, a monitorear el tiempo de desarrollo y a entender el entorno de los sistemas. Una de las grandes ventajas de Rose es que utiliza la notación estándar en la arquitectura de software(UML), la cual permite a los arquitectos de software y desarrolladores visualizar el sistema completo utilizando un lenguaje común, además los diseñadores pueden modelar sus componentes e interfaces en forma individual y luego unirlos con otros componentes del proyecto.³³

Rational Rose permite completar una gran parte de las disciplinas (flujos fundamentales) del proceso unificado de Rational (RUP):

- Modelado del negocio
- Captura de requisitos (parcial)
- Análisis y diseño (completo)
- Implementación (como ayuda)
- Control de cambios y gestión de configuración (parte)

Características principales:

- Admite como notaciones: UML, COM, OMT y Booch
- Realiza chequeo semántico de los modelos
- Ingeniería “de ida y vuelta”: Rose permite generar código a partir de modelos y viceversa
- Desarrollo multiusuario

³³ Ugaz, Max. Disponible en <http://www.usmp.edu.pe/publicaciones/boletin/fia/info36/proyectos.html>

- Integración con modelado de datos
- Generación de documentación
- Tiene un lenguaje de script para poder ampliar su funcionalidad

1.9 Tecnologías para el desarrollo del Simulador

Antes de hacer un análisis de las herramientas y lenguajes que se utilizarán para la implementación del simulador se debe decir que el software finalmente estará accesible desde un mapa conceptual, realizado por la Msc. Lydia Rosa Ríos. Partiendo del hecho de que el mapa conceptual se encontrará accesible desde una página Web, se hace necesario que el simulador sea un software que se pueda ejecutar desde una página Web, sin necesidad de descargarlo.

1.9.2 Mapas Conceptuales

Los mapas conceptuales ofrecen ventajas como herramientas para la navegación en Internet, aprovechando la flexibilidad que ofrecen las nuevas tecnologías. El problema de la navegación, común en los sistemas de multimedia y persistente en la navegación de la WWW, se resuelve fácil con estos mapas. Su menú de iconos corresponden a diversos medios (texto, imágenes, vídeos, otros mapas conceptuales, simulaciones, etc.) y éstos facilitan el proceso de navegación, ya que puede accederse a ellos desde cualquier nodo de la red.³⁴

Los mapas conceptuales resultan de gran utilidad en la educación a distancia y, son también accesibles como páginas de HTML. Por eso, los sistemas insertados en sus iconos deben ser fáciles de descargar.

1.9.3 World Wide Web

³⁴ Cañas, Alberto J. , Ford, Kenneth. Disponible en <http://www.ihmc.us/users/acanas/Publications/RevistaInformaticaEducativa/HerramientasConsConRIE.htm>

Llamada así por sus siglas WWW, es la aplicación que ha popularizado a Internet, y ha sido la causante de la explosión de la red de redes.

Hablando sencillamente, la WWW basa su existencia en dos aplicaciones: los servidores de Web y los clientes. Los primeros son programas dedicados, instalados en las máquinas nodos de la red, que atienden a todos los pedidos de los clientes. Estos solicitan archivos, recursos de la Web, en general, archivos HTML y gráficos. Los archivos HTML son archivos de texto, que contienen una descripción de lo que vemos como página en nuestro browser. La sigla HTML significa Hypertext Markup Language (lenguaje de marcación de hipertexto), y es el lenguaje que describe las páginas Web.

En esas páginas se colocan textos, gráficos y otros recursos. Es trabajo del cliente, nuestro browser, interpretar esa página, mostrarla en pantalla, y reclamar todo lo que se necesite para armarla, como los gráficos y archivos de sonido que la pueden acompañar.³⁵

1.9.4 Java. Applet

Java es un lenguaje de programación orientado a objetos desarrollado por James Gosling y sus compañeros de Sun Microsystems al inicio de la década de 1990. Y ofrece toda la funcionalidad de un lenguaje potente, pero sin las características menos usadas y más confusas de éstos. Java elimina muchas de las características de otros lenguajes como C++, para mantener reducidas las especificaciones del lenguaje y añadir características muy útiles como el garbage collector (reciclador de memoria dinámica).³⁶

Los programas Java son, al fin, otro recurso más, como pueden serlo una página HTML o un archivo gráfico. Al igual que estos últimos viajan desde el servidor a la máquina cliente, y se ejecutan en ésta.

³⁵ Apuntes de Java. Disponible en <http://www.monografias.com/trabajos16/java/java.shtml#browsers>

Los programas desarrollados en Java presentan diversas ventajas frente a los desarrollados en otros lenguajes como C/C++. La ejecución de programas en Java tiene muchas posibilidades: ejecución como aplicación independiente, ejecución como applet, ejecución como servlet, etc. Un applet es una aplicación especial que se ejecuta dentro de un navegador o browser (por ejemplo Netscape Navigator o Internet Explorer) al cargar una página HTML desde un servidor Web. El applet se descarga desde el servidor y no requiere instalación en el ordenador donde se encuentra el browser. Un servlet es una aplicación sin interface gráfica que se ejecuta en un servidor de Internet. La ejecución como aplicación independiente es análoga a los programas desarrollados con otros lenguajes.³⁷

Características

- Java es un lenguaje orientado a objetos: Esto es lo que facilita abordar la resolución de cualquier tipo de problema.
- Es un lenguaje sencillo, aunque sin duda potente.
- La ejecución del código Java es segura y fiable: Los programas no acceden directamente a la memoria del ordenador, siendo imposible que un programa escrito en Java pueda acceder a los recursos del ordenador sin que esta operación le sea permitida de forma explícita. De este modo, los datos del usuario quedan a salvo de la existencia de virus escritos en Java. La ejecución segura y controlada del código Java es una característica única, que no puede encontrarse en ninguna otra tecnología.
- Es totalmente multiplataforma: Es un lenguaje sencillo, por lo que el entorno necesario para su ejecución es de pequeño tamaño y puede adaptarse incluso al interior de un navegador.³⁸

³⁶ Lenguaje de programación Java. Disponible en http://es.wikipedia.org/wiki/Lenguaje_de_programaci%C3%B3n_Java

³⁷ Introducción a Java J2SE. Disponible en <http://diario.de/medianero.script/introduccion-a-java-j2se.html>

³⁸ García Peñalvo, Francisco José. Disponible en http://pisuerga.inf.ubu.es/lsi/Invest/Java/Tuto/l_2.htm

Las applets (miniaplicación) son programas escritos en Java que sirven para "dar vida" a las páginas Web (interacción en tiempo real, inclusión de animaciones, sonidos...), de ahí su potencia.

Las applets son programas que se incluyen en las páginas Web son muy seguros, pues en general no permiten realizar cualquier acción que pudiera dañar la máquina o violar la intimidad del que visita la página Web y no necesitan ser descargados.

Las applets son ejecutadas en la máquina cliente, con lo que no existen ralentizaciones por la saturación del módem o del ancho de banda. Permiten cargar a través de la red una aplicación portable que se ejecuta en el navegador. Para que esto ocurra tan sólo hace falta que el navegador sea capaz de interpretar Java.

A las páginas que contienen applets se las denomina páginas Java-Powered. Las applets pueden ser visualizadas con la herramienta appletviewer, incluido en el JDK de Java.³⁹

Los programas escritos en Java pueden ser ejecutados en cualquier plataforma computacional, el lenguaje está muy orientado al trabajo en red, soportando protocolos como TCP/IP, UDP, HTTP y FTP.

Dada las características que debe tener el simulador se confeccionará usando applets. La construcción de estos applets puede ser sencilla o compleja, dependiendo del problema a resolver. Existen diferentes herramientas para crear los applets, entre ellas una herramienta llamada Ejs (Easy Java Simulations) o en cualquier entorno de desarrollo de Java (NetBeans, Eclipse).

Los Applets de Java constituyen la mejor opción para realizar la simulación; al reunir todas las condiciones que se necesitan para ubicar los mismos dentro de los nodos de los mapas conceptuales.

1.9.5 Easy Java Simulations

³⁹ Manzanedo del Campo, Miguel Ángel. Disponible en http://pisuerga.inf.ubu.es/lsi/Invest/Java/Tuto/VI_1.htm

Easy Java Simulations es una herramienta muy buena para hacer applets y para crear simulaciones en particular. Ejs, es un entorno gráfico de programación orientado al desarrollo de simulaciones científicas y técnicas. Permite desarrollar simulaciones Java a personas no expertas en informática y desconocedoras de éste lenguaje.⁴⁰

Pero 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.

1.9.6 Eclipse

Eclipse es una plataforma de desarrollo open source basada en Java. Es un desarrollo de IBM cuyo código fuente fue puesto a disposición de los usuarios. En sí mismo Eclipse es un marco y un conjunto de servicios para construir un entorno de desarrollo a partir de componentes conectados (plug-in). Hay plug-ins para el desarrollo de Java (JDT Java Development Tools) así como para el desarrollo en C/C++, COBOL, etc.⁴¹

Eclipse es un Software de Desarrollo Open-Source y multiplataforma que proporciona robustez, amplias características, calidad comercial y una plataforma industrial para el desarrollo de aplicaciones de alta calidad. Según la descripción de su página oficial, "Eclipse es un IDE (Integrated Development Environment) para cualquier cosa y ninguna en particular". Eclipse es, en el fondo, un armazón (workbench) de desarrollo sobre el que se pueden montar herramientas de desarrollo para cualquier lenguaje, mediante la implementación de los plugins adecuados.⁴²

⁴⁰ Esquembre, Francisco. Disponible en http://www.um.es/fem/Ejs/Ejs_es/index.html

⁴¹ Gutiérrez, Juan. Disponible en http://www.uv.es/~jgutierr/MySQL_Java/TutorialEclipse.pdf

⁴² Eclipse. Disponible en <http://ame.endesa.es/confluence/display/AMEDev/Eclipse>

Herramienta para el programador desarrollada principalmente para el desarrollo de aplicaciones Java, facilitando al máximo la gestión de proyectos colaborativos mediante el control de versiones 'cvs', es posible también con subversión, exportar e importar proyectos.⁴³

Eclipse fue desarrollado originalmente por IBM como el sucesor de su familia de herramientas para VisualAge. Eclipse es ahora desarrollado por la Fundación Eclipse, una organización independiente sin ánimo de lucro que fomenta una comunidad de código abierto y un conjunto de productos complementarios, capacidades y servicios.⁴⁴

1.9.7 NetBeans

NetBeans es un entorno integrado de desarrollo (IDE) de código abierto cuya principal misión es hacer fácil el desarrollo de todo tipo de aplicaciones en la plataforma JAVA (JVM). Con NetBeans es posible desarrollar tanto aplicaciones de escritorio, como aplicaciones empresariales en varias capas, o programas para todo tipo de dispositivos móviles.⁴⁵

Con el anuncio de la versión 5.5, NetBeans IDE se superan las deficiencias asociadas con otras plataformas de desarrollo y se abren nuevas posibilidades para alcanzar el desarrollo rápido y eficiente de aplicaciones multiplataforma. En su núcleo, NetBeans IDE es una herramienta de desarrollo Java, escrita puramente sobre la base de la tecnología Java, de modo que puede ejecutarse en cualquier ambiente que ejecute Java, lo cual, por supuesto, es casi en todas partes.

NetBeans es un producto de código abierto, con todos los beneficios del software disponible en forma gratuita, el cual ha sido examinado por una comunidad de desarrolladores. Este enfoque de bienes comunes creativos ha permitido una mayor capacidad de uso, con cada nueva versión, y ha proporcionado a los desarrolladores

⁴³ Eclipse SDK. Disponible en http://campusvirtual.unex.es/cala/epistemowikia/index.php?title=Eclipse_SDK

⁴⁴ Eclipse (software). Disponible en [http://es.wikipedia.org/wiki/Eclipse_\(software\)](http://es.wikipedia.org/wiki/Eclipse_(software))

⁴⁵ López, Ignacio. Disponible en <http://ame.endesa.es/confluence/display/AMEDev/NetBeans>

mayor flexibilidad, al modificar el IDE, si así lo desean. Para la mayoría de los desarrolladores de aplicaciones, el portal de desarrollo netbeans.org es la dirección integrada para descargas del ambiente, conjuntamente con paquetes de características especiales e información sobre el IDE.

De igual forma, NetBeans ofrece las ventajas de que Sun Microsystems lo ha creado, respaldado y ha abierto su código. La relación de Sun con el código básico de NetBeans se remonta a ocho años atrás aproximadamente y, dado que el IDE tenía el código abierto, en el año 2000, la compañía ha seguido desempeñando el papel de patrocinadora, lo cual garantiza que NetBeans es un ambiente de desarrollo empresarial con soporte total.

Aparte de la filosofía de distribución y desarrollo que respalda a NetBeans, el IDE ofrece a los desarrolladores numerosas ventajas, en la creación de nuevas aplicaciones multiplataforma. En una era en la cual la arquitectura orientada al servicio (SOA) requiere servicios con cierta relación que manejen procesos específicos del negocio, NetBeans satisface los requisitos con conjuntos de herramientas independientes de la plataforma, modulares y orientadas al objeto.

La facilidad de uso de NetBeans también resulta llamativa, ya que no es necesario ir más allá del portal netbeans.org, para buscar las características adicionales y los “plugins” de los paquetes de funciones que abarcan una diversidad de requerimientos — desde C/C++ hasta movilidad y la Web. Una vez que se descarga, NetBeans es fácil de utilizar, gracias a su interfaz de usuario y a funciones como la recién anunciada Visual Web Pack (VWP). Con VWP, los componentes JavaServer Faces pueden arrastrarse y soltarse para establecer propiedades y crear códigos para los manipuladores de eventos a nivel del servidor.

Adicionalmente, con el configurador NetBeans Swing GUI, tanto los usuarios finales como los desarrolladores obtienen mayor facilidad de uso. Swing simplifica drásticamente la creación de interfaces gráficas de usuario para grandes aplicaciones

de Internet y cliente, además de permitir a los desarrolladores manejar diferentes guías de estilo en diversas plataformas y garantizar la ubicación de aplicaciones en una amplia variedad de lenguajes humanos.

Las características de NetBeans de flexibilidad entre plataformas, el cumplimiento de UML y la capacidad de administrar la complejidad ayudan a garantizar que las aplicaciones cumplan con los requerimientos específicos del negocio.

La nueva velocidad de desarrollo requiere también la visibilidad y confiabilidad del código abierto, conjuntamente con las capacidades de Java que sólo Sun puede proporcionar. Gracias a NetBeans, los desarrolladores pueden confiar en una plataforma de desarrollo que integra todas las piezas críticas, necesarias en cada nivel de creación de aplicaciones.⁴⁶

1.9.8 Macromedia Flash

Macromedia Flash es una herramienta de creación de páginas Web que permite crear sitios ricos en gráficos y animaciones con más control sobre estos gráficos. Los gráficos y animaciones creados con Flash son mucho más pequeños que los formatos tradicionales (gifs animados, avi, etc) porque utilizan vectores en vez de mapas de bits. Las animaciones hechas en flash se cargan en los navegadores Web mas conocidos.

⁴⁷

Flash es casi con total seguridad el mejor programa de animación en 2D, permite mucha creatividad en el diseño, tiene unas herramientas y una interfaz que lo hacen muy fácil de usar, se comunica fácilmente con los lenguajes con los que coexiste, cuenta con una amplia lista de componentes prefabricados, tiene un buen soporte de conexiones permanentes con XML Sockets y Flash Communication Server, un buen lenguaje orientado a objetos con el ActionScript 2.0.⁴⁸

⁴⁶ Desarrollo de aplicaciones multiplataforma con NetBeans IDE. Disponible en http://www.sun.com/emrkt/innercircle/newsletter/latam/0207latam_feature.html

⁴⁷ Sve, Julian. Disponible en http://www.svetlian.com/Webmaster/flash_tutor1.htm

⁴⁸ Martí García, Joaquín. Disponible en http://www.wikilearning.com/puntos_fuertes_de_flash-wkccp-69-4.htm

1.9.9 W-Prolog

W-PROLOG es un interpretador implementado en JAVA para un lenguaje como PROLOG. W-PROLOG creado por Michael Winikkoff tiene una agradable interfaz para usuarios mejor que los sistemas prolog.

El sistema W-Prolog es pequeño y es comparativamente simple. Sin embargo esto no lo hace particularmente rápido.

Puede ser ejecutado como un applet o como una aplicación, también puede ser llamado desde otro programa.

W-Prolog es actualmente usado como parte de la Guía para programación en prolog de Roman Barták en el centro de antropología y de Computación de la Universidad de Kent.⁴⁹

⁴⁹ Barrera Cerna, Manuel Iván. Disponible en <http://html.rincondelvago.com/aplicaciones-de-prolog-en-java.html>

Capítulo 2 Análisis y diseño de la propuesta de solución

Introducción.

Para el diseño e implementación de un simulador que apoye el aprendizaje del proceso de unificación se necesita hacer un análisis detallado del funcionamiento del contexto donde se desarrolló el estudio y un diagnóstico de cómo se desarrolla esta actividad.

En este capítulo el análisis del sistema, utilizando para su modelado el Lenguaje Unificado de Modelación (UML), que permite representar el diagrama de casos de uso del sistema y la especificación de los mismos con una pantalla de interfaz correspondiente.

2.1 Caracterización sobre aspectos generales de la estructura, organización y funcionamiento del contexto donde se desarrolló el estudio.

Conocer el contexto donde tendrá lugar cualquier tipo de estudio científico, constituye un elemento importante para el investigador, pues ayuda a entender mejor el comportamiento del problema y en correspondencia con esto permite diseñar mejor las propuestas de solución. En este caso se tiene como objeto de estudio al proceso enseñanza -aprendizaje de la asignatura programación Lógica, de la carrera Licenciatura en Ciencia de la Computación de la Facultad Matemática Física y Computación de la Universidad Central "Martha Abreu" de Las Villas.

La Facultad de Matemática-Física-Computación comenzó en 1955 y en la actualidad atiende las carreras Licenciatura en Ciencias de la Computación y Licenciatura en Matemática. Tiene adjunto el Centro de Estudios Informáticos (CEI). El postgrado incluye: Doctorado en Ciencias Técnicas, Maestría en Computación Aplicada, Maestría en Matemática Aplicada y Especialidad en Base de Datos. Ocupa un lugar destacado en la investigación y en la colaboración con el extranjero.

La carrera de Ciencias de la Computación tiene una duración de 5 años (10 semestres), donde se imparten un total de 58 asignaturas organizadas en 17 disciplinas.

El nuevo Plan de Estudios se extiende durante cinco años y contempla las siguientes disciplinas con sus correspondientes asignaturas Análisis Matemático, Ecuaciones Diferenciales, Álgebra, Programación, Estructura de Datos y Algoritmos, Lenguajes de Programación, Ingeniería de Software, Matemática Computacional, Lógica, Matemática Discreta, Análisis y Diseño de Algoritmos, Compilación, Sistemas de Bases de Datos, Sistemas de Información, Inteligencia Artificial, Programación Lógica, Simulación, Historia de la Computación, Sistemas de Operación y Redes, Sistemas de Operación, Programación de Máquinas: Estudio de las principales arquitecturas de computadoras y de la programación a bajo nivel, Modelos de Optimización, Probabilidad y Estadística, Matemática Numérica, entre otras.⁵⁰

2.2 Diagnóstico inicial del problema para desarrollar la propuesta de diseño del Simulador que mejore la comprensión del proceso de matching.

Para propiciar el diseño de este software educativo se realizó un diagnóstico en los alumnos de 4to año de Licenciatura en Ciencia de la Computación con el objetivo de conocer la dificultad que tienen para acceder a los servicios de información científica, tecnológica, corporativa y de dirección para apoyar bibliográficamente el proceso de enseñanza-aprendizaje. La población correspondió al total de los estudiantes matriculados en dicha carrera para el curso 2006 – 2007. La selección de la muestra correspondió con el 100 % de la población. El muestreo fue de tipo intencional.

Para aplicar la encuesta se realizó una breve explicación a los estudiantes sobre el objetivo general de nuestra investigación, solicitándoles su cooperación. Se confeccionó la encuesta (ver Anexo No 1) la cual tenía la intención de caracterizar el aprendizaje del proceso de matching de la asignatura Programación Lógica.

⁵⁰ Colectivo de autores. Disponible en <http://www.uclv.edu.cu/uclv/Facultades/mat/?queryitem=Ciencia%20de%20la%20Computación>

Pregunta 1: Evalúa el interés de los estudiantes en la asignatura anteriormente. De los 36 encuestados, el 14 % conocía la asignatura y se habían interesado en la misma, el restante 86 % no tenían conocimiento de la misma. De aquí se infiere que aún no existe buena divulgación acerca del contenido de la asignatura.

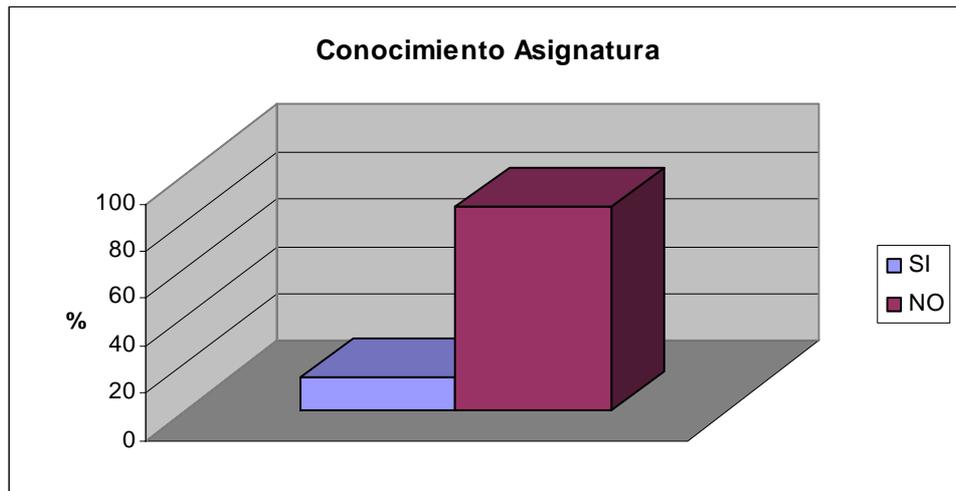


Figura 2.1 Gráfico sobre el conocimiento de la asignatura.

Pregunta 2: Evalúa la dificultad con el estudio de la asignatura. De los 36 estudiantes, el 88.9 % expresó que la asignatura era difícil, el 11.1 % manejable y ninguno fácil. De aquí se deriva que para los estudiantes esta es una asignatura con un alto grado de dificultad.

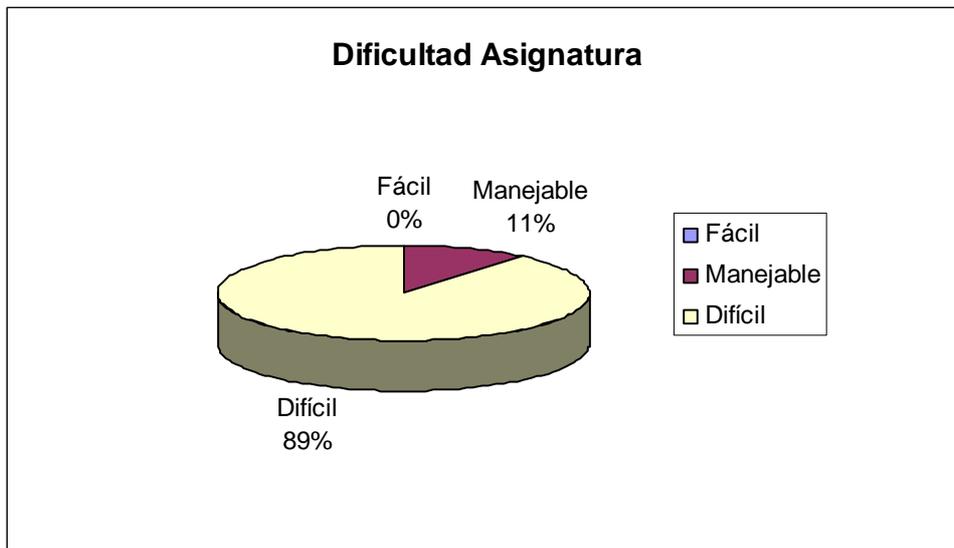


Figura 2.2 Gráfico sobre la dificultad de la asignatura.

Pregunta 3: Evalúa la disponibilidad de bibliografía. De los 36 alumnos, 50 % considera que la bibliografía sobre la programación lógica es escasa, 44.4% que no es muy abundante y 5.5 % cree que es abundante. De aquí podemos inferir que existe un déficit de bibliografía sobre el tema.

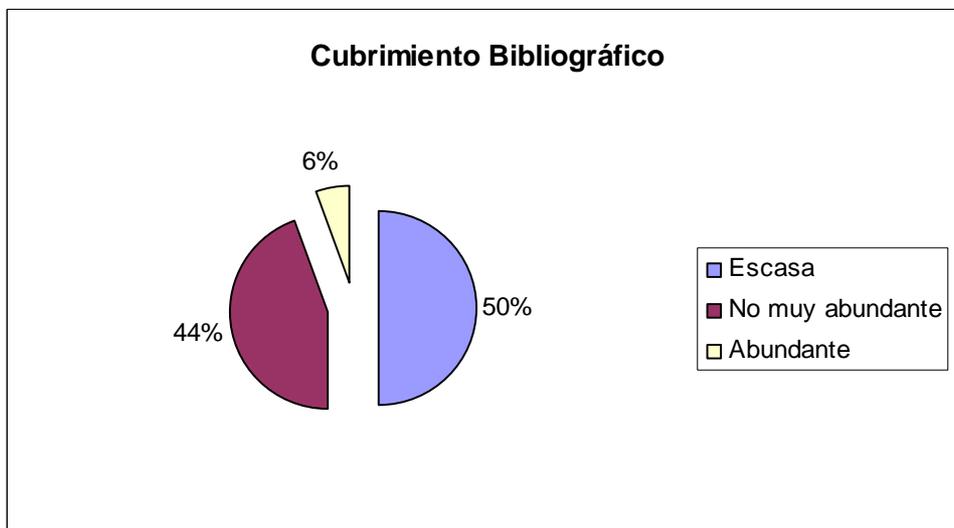


Figura 2.3 Gráfico acerca del cubrimiento bibliográfico de la asignatura.

Pregunta 4: Evalúa la comprensión del proceso de matching, De los 36 alumnos, el 6

% considera que es fácil, 25 % normal y 25 69.4 % difícil. De aquí se deriva que este proceso es difícil para la mayoría de los estudiantes.

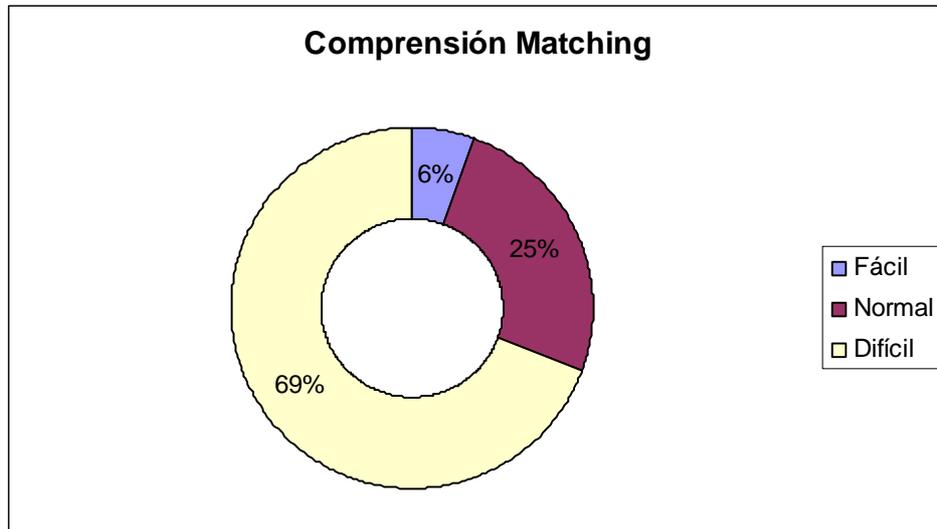


Figura 2.4 Gráfico acerca de la comprensión del proceso de matching.

Pregunta 5: Se evalúa la orientación acerca del estudio de la asignatura. De los 36 encuestados, 39 % plantea que es buena, 47 % regular y 14 % insuficiente. De este resultado se deduce que las orientaciones no cubren las expectativas de los estudiantes en su totalidad y que son necesarias otras herramientas.

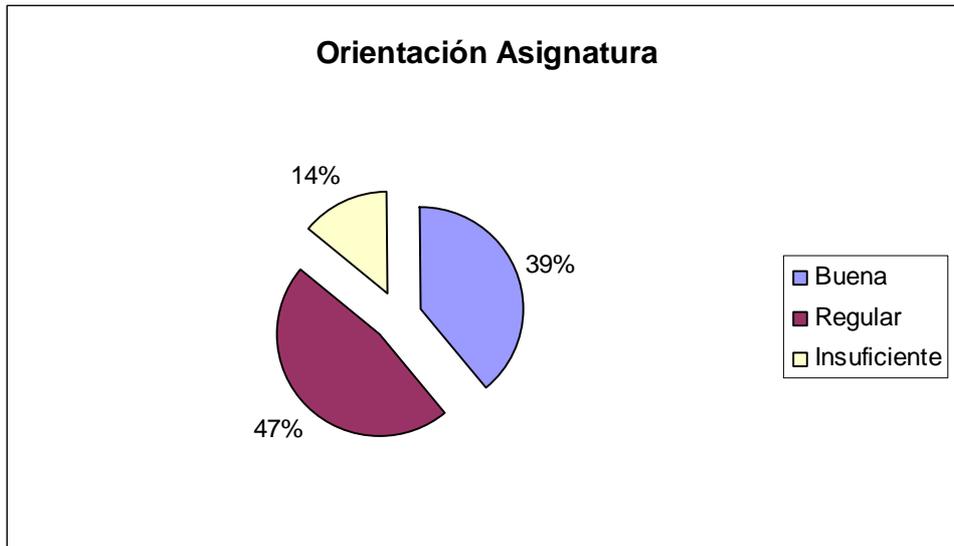


Figura 2.5 Gráfico acerca de la orientación de la asignatura.

Pregunta 6: Evalúa el conocimiento de algún recurso informático accesible para comprender mejor el proceso de matching, de los 36 alumnos, el 75 % plantea no conocer ningún recurso informático que les brinde esta posibilidad, 25 % expresaron lo contrario. De este resultado se deduce que la mayoría de los estudiantes no tienen conocimiento sobre la existencia de algún recurso informático que les ofrezca la posibilidad de comprender mejor el proceso de matching

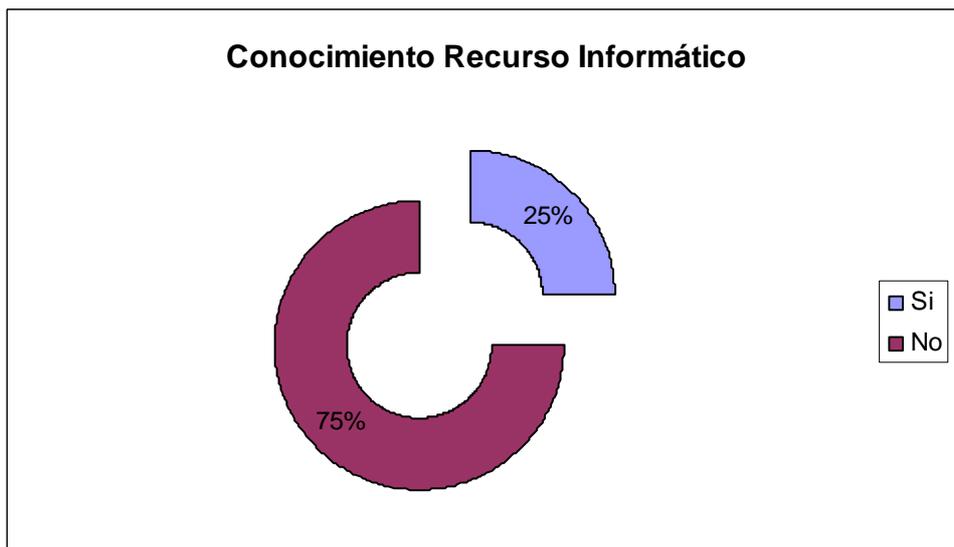


Figura 2.6 Gráfico sobre el conocimiento de un recurso informático para el matching.

Pregunta 7: de los 9 estudiantes que plantearon conocer un recurso informático que les brindaba la posibilidad de comprender el proceso de matching, 6 (%) refirió que utilizan foros de discusión en Internet, 2 (%) afirmaron que han interactuado con algún tutorial inteligente, sólo 1 (%) con un entrenador y ninguno conoce ningún simulador a través del cual puede comprender mejor el proceso de matching en lenguaje Prolog. De aquí se deriva que la mayoría de los estudiantes no ha interactuado con ningún simulador que les permita comprender mejor el proceso de matching en lenguaje Prolog, por lo que queda plenamente justificada nuestra intervención en esta investigación.

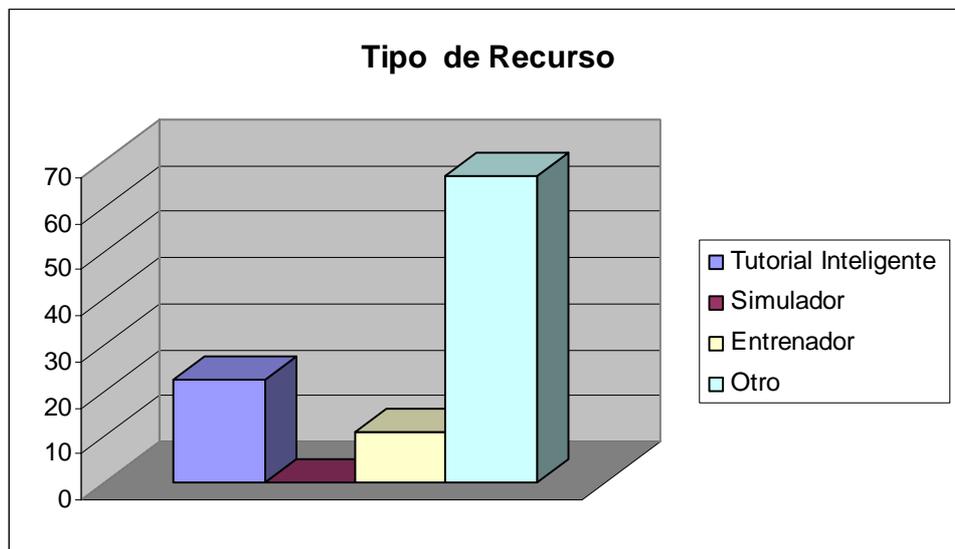


Figura 2.7 Gráfico sobre el tipo de recurso informático para el matching.

2.3 Descripción del negocio.

El diseño del simulador se realiza usando la metodología RUP explicada en el Capítulo 1 de esta investigación.

2.3.1 Representación de los casos de uso del negocio.

A continuación se representan los casos de uso del negocio, indicando quiénes son los actores y trabajadores del negocio y por qué lo son. Un modelo de casos de uso del negocio describe los procesos de negocio de una empresa en términos de casos de uso del negocio y actores del negocio que se corresponden con los procesos del negocio y los clientes, respectivamente. Al igual que el modelo de casos de uso para un sistema software, el modelo de casos de uso del negocio presenta un sistema (en este caso, el negocio) desde la perspectiva de su uso, y esquematiza cómo proporciona valor a sus usuarios (en este caso, sus clientes y socios).⁵¹

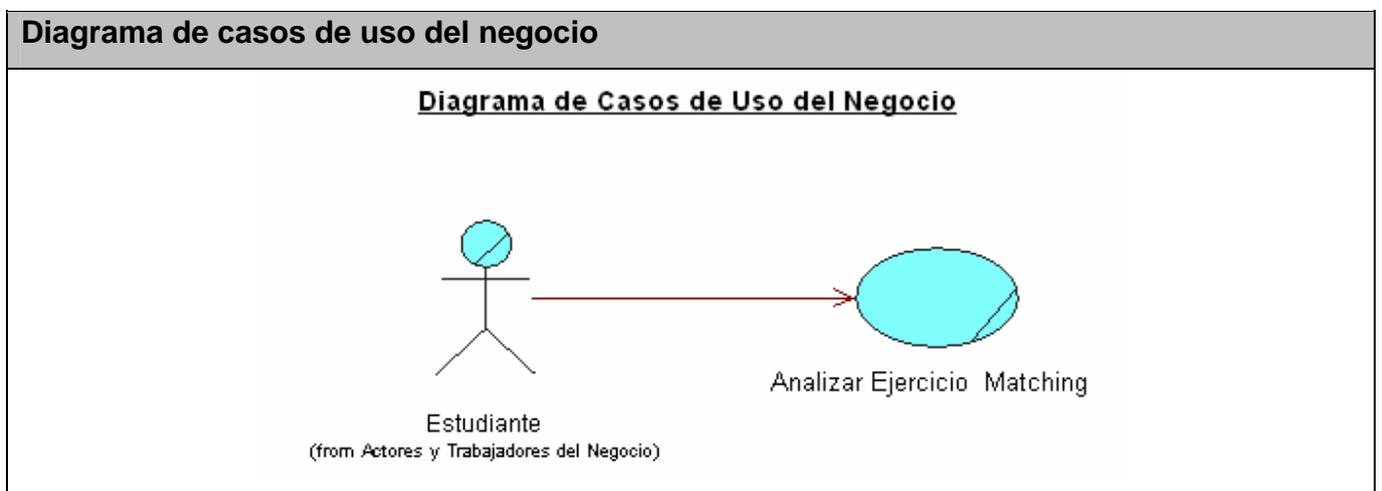


Figura. 2.1 Diagrama de casos de uso del negocio.

2.3.2 Actores y trabajadores del negocio.

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. Un trabajador del negocio es una abstracción de una persona, una máquina o un sistema automatizado; que actúa en el negocio realizando una o varias

⁵¹ Jacobson, Ivar, Booch, Grady; 2000.

actividades, interactuando con otros trabajadores del negocio y manipulando entidades del negocio. Representa un rol.⁵²

| Actores | Justificación |
|------------|--|
| Estudiante | Analizar los ejercicios y comprender el funcionamiento del proceso de Matching de la Programación Lógica |

Tabla 2.1 Descripción de los actores del negocio.

| Trabajadores | Justificación |
|--------------|---|
| Profesor | Explica como funciona el proceso de matching y de simular el funcionamiento del prolog cuando realiza este proceso. |

Tabla 2.2 Descripción de los trabajadores del negocio.

2.3.3 Caso de uso expandido.

| | |
|---|---|
| CASO DE USO DEL NEGOCIO (1) | ANALIZAR EJERCICIO MATCHING |
| Actores: | Estudiante |
| Propósito | Que el estudiante entienda el funcionamiento del proceso de Matching de la Programación Lógica. |
| RESUMEN: El caso de uso se inicia cuando el profesor presenta el ejercicio a los estudiantes, auxiliándose de un documento o libro. El profesor resuelve el ejercicio y pregunta si entienden como funciona el proceso de Matching. Si los estudiantes no entienden el profesor hace una explicación paso a paso y simula como funciona el prolog internamente para que el estudiante pueda entender. El caso de Uso finaliza cuando el Estudiante llega a entender como funciona este proceso. | |
| ACCIÓN DEL ACTOR | RESPUESTA DEL PROCESO DE NEGOCIO |

⁵² González, Anaisa. Modelamiento del Negocio.

| | | | |
|---------------------|---|---|---|
| | | 1 | El Profesor presenta el Ejercicio de Matching. |
| | | 2 | Si el estudiante no entiende como funciona el proceso en la solución del Ejercicio. |
| | | 3 | El Profesor hace una explicación paso a paso del proceso. |
| | | 4 | Simula el funcionamiento del Prolog cuando hace un matcheo. |
| 6 | Cuando el estudiante comprende el funcionamiento del proceso de Matching finaliza el caso de uso. | | |
| 2) PRIORIDAD | | | |

Tabla 2.3 Descripción del caso de uso Analizar Ejercicio Matching.

2.3.4 Diagrama de actividades.

Un diagrama de actividad es un caso especial de un diagrama de estado en el cual todos (o la mayoría) de los estados son estados de acción y en el cual todas (o la mayoría) de las transiciones son disparadas para el completamiento de la acción, en el estado fuente.

El diagrama de actividad completo es asociado (a través del modelo) a una clase o a la implementación de una operación o un caso de uso de proceso. El propósito de este diagrama es enfocar en los flujos manipulados por el procesamiento (como oposición a eventos externos).⁵³

⁵³ Jacobson, Ivar, Booch, Grady; 2000.

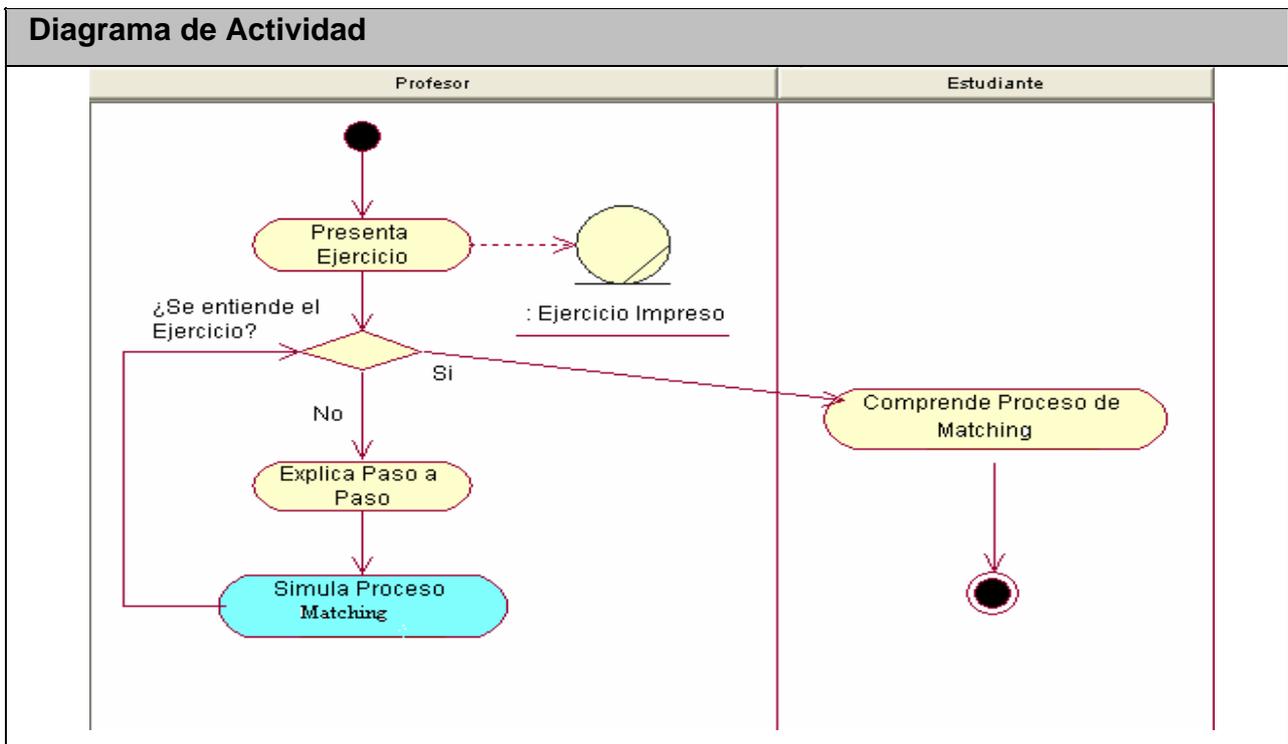


Figura. 2.2 Diagrama de Actividad.

2.3.5 Diagrama de clases del modelo de objetos.

El diagrama de clases del modelo de objeto, es un 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.⁵⁴

⁵⁴ Jacobson, Ivar, Booch, Grady; 2000.

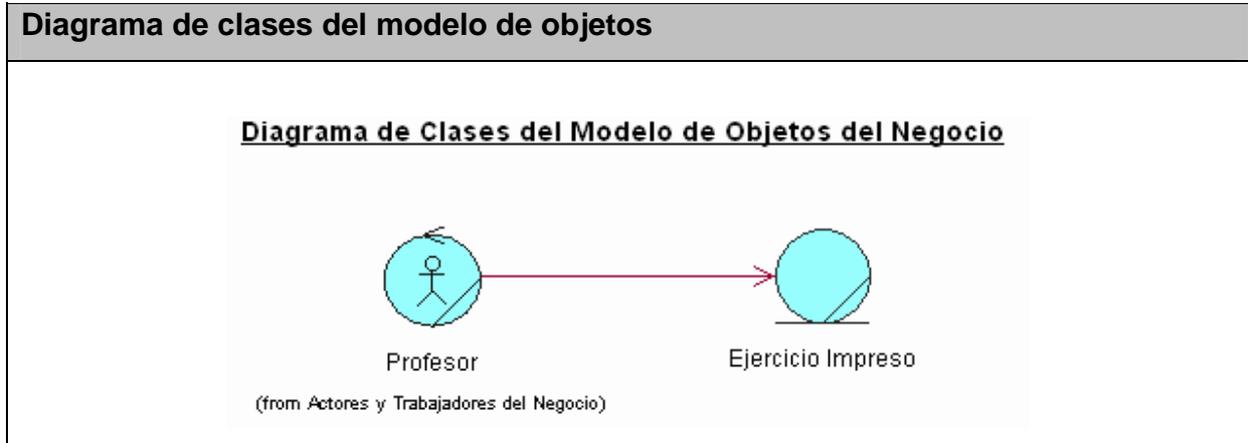


Figura. 2.3 Diagrama de clases del modelo de objetos.

2.4 Especificación de los requisitos de software.

Para determinar los requerimientos funcionales del sistema. Se hizo una entrevista al cliente del software y que a su vez es un experto en el tema (Ver Anexo 2). De allí se pudo arribar a las siguientes conclusiones:

2.5.1 Listado de los requerimientos funcionales.

1. Mostrar enunciado de los ejemplos.
2. Mostrar explicación de cada uno de los ejemplos
3. Simular los ejemplos.
 - 3.1 Ejecutar los ejemplos paso a paso
 - 3.2 Mostrar Historial del ejemplo
 - 3.3 Mostrar Variables e instancias en cada caso
4. Detener ejecución de los ejemplos.

2.5.2 Listado de los requerimientos no funcionales.

Los requerimientos no funcionales son propiedades o cualidades que el producto debe tener. Debe pensarse en estas propiedades como las características que hacen al

producto atractivo, usable, rápido o confiable.

Los requerimientos no funcionales incluyen:

Conjunto de facilidades.

Capacidades.

Seguridad.

Requerimientos de apariencia o interfaz externa

El software debe brindar una interfaz simple y de fácil uso para que el usuario no tenga dificultad al interactuar con el sistema. La interfaz del sistema se ajustará a los estándares establecidos para el desarrollo de un buen diseño. Deberá tener consistencia con el mundo real de manera que los conceptos manejados sean conocidos y familiares por los usuarios, para que les sea fácil su uso y aprendizaje. Estará diseñada de modo tal que el usuario pueda ir de un punto a otro dentro de ella con gran facilidad.

Requerimientos de Usabilidad

La herramienta será utilizada por cualquier persona que tenga acceso el sitio y que posea conocimientos básicos en el manejo de la computadora y de un ambiente Web en sentido general.

Requerimientos de Rendimiento

La herramienta propuesta debe ser rápida y el tiempo de respuesta debe ser el mínimo posible, adecuado a la rapidez con que el cliente requiere la respuesta a su acción.

Requerimientos de Portabilidad

La herramienta propuesta podrá ser usada sobre el Sistema Operativo Windows 98 o superior o Linux.

Requerimientos de Software

Los clientes tendrán acceso a través de cualquier navegador Web. Recomendados: Internet Explorer 4.0 o cualquier otro navegador.

Requerimientos de Hardware

Se requiere computadoras conectadas a una red y tener al menos 64 MB de RAM.

Restricciones en el diseño y la implementación

Se utilizarán herramientas de desarrollo que garanticen la calidad de todo el ciclo de desarrollo del producto.

2.5 Descripción del sistema propuesto.

2.5.1 Actores del sistema.

Un actor no es parte del sistema, es un rol de un usuario que puede intercambiar información o puede ser un recipiente pasivo de información y representa a un ser humano, a un software o a una máquina que interactúa con el sistema.⁵⁵

A continuación se definen los actores del sistema propuesto:

| Actores | Justificación |
|---------|---------------|
|---------|---------------|

⁵⁵ Jacobson, Ivar, Booch, Grady; 2000.

| | |
|------------|--|
| Estudiante | Visualiza información y ejecuta la simulación de los ejemplos hasta entender el proceso de Matching. |
|------------|--|

Tabla 2.4 Descripción de los actores del sistema.

2.5.2 Diagrama de casos de usos del sistema.

Cada forma en que los actores usan el sistema se representa con un Caso de Uso. Los Casos de Uso son fragmentos de funcionalidad que el sistema ofrece para aportar un resultado de valor para sus actores. De manera más precisa, un Caso de Uso especifica una secuencia de acciones que el sistema puede llevar a cabo interactuando con sus actores, incluyendo alternativas dentro de la secuencia.

Un Modelo de Casos de Uso es un modelo del sistema que contienen actores, Casos de Uso y relaciones.⁵⁶

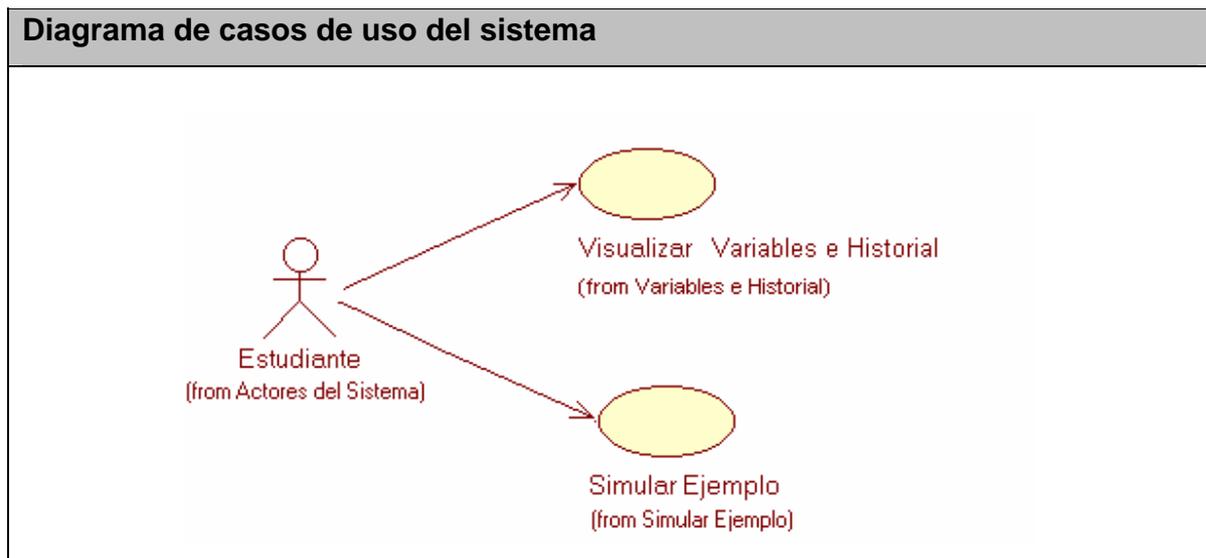


Figura. 2.4 Diagrama de casos de uso del sistema.

2.5.3 Descripción de los casos de uso del sistema.

A través de las tablas siguientes se enuncian y describen los casos de uso del sistema y se referencian con los requerimientos funcionales planteados.

| | |
|--------------------|---|
| CU-1 | <i>Visualizar Variables e Historial</i> |
| Actores: | Estudiante |
| Propósito: | Poder ver cuales son las variables del ejemplo y los valores que toman en cada momento. Puede ver también el historial de se v ejecutando el ejemplo. |
| Referencia: | RF1, RF2 |
| Resumen: | Se selecciona la opción de Variables o de Historial de los Ejemplos. En cualquier de los casos se muestra una tabla con la información solicitada. |

| | |
|--------------------|---|
| CU-2 | <i>Simular Ejemplo</i> |
| Actores: | Estudiante |
| Propósito: | Ver la simulación de cada ejemplo. |
| Referencia: | RF3 |
| Resumen: | Se muestra la traza del ejemplo como si estuviera corriendo en Prolog, con la diferencia que el usuario tiene la posibilidad de ejecutar paso a paso el ejemplo. Para cada ejemplo se puede leer una explicación del mismo. |

2.5.4 Diagrama de clases.

En estos diagramas es donde se pueden ver las relaciones entre todas las clases del sistema, además de los atributos y operaciones de cada una de las mismas.

El Diagrama de Clases es el diagrama principal para el análisis y diseño. Un diagrama de clases presenta las clases del sistema con sus relaciones estructurales y de herencia. La definición de clase incluye definiciones para atributos y operaciones. El

⁵⁶ Jacobson, Ivar, Booch, Grady; 2000.

modelo de casos de uso aporta información para establecer las clases, objetos, atributos y operaciones.⁵⁷

⁵⁷ Jacobson, Ivar, Booch, Grady; 2000.

Diagrama de clases del sistema

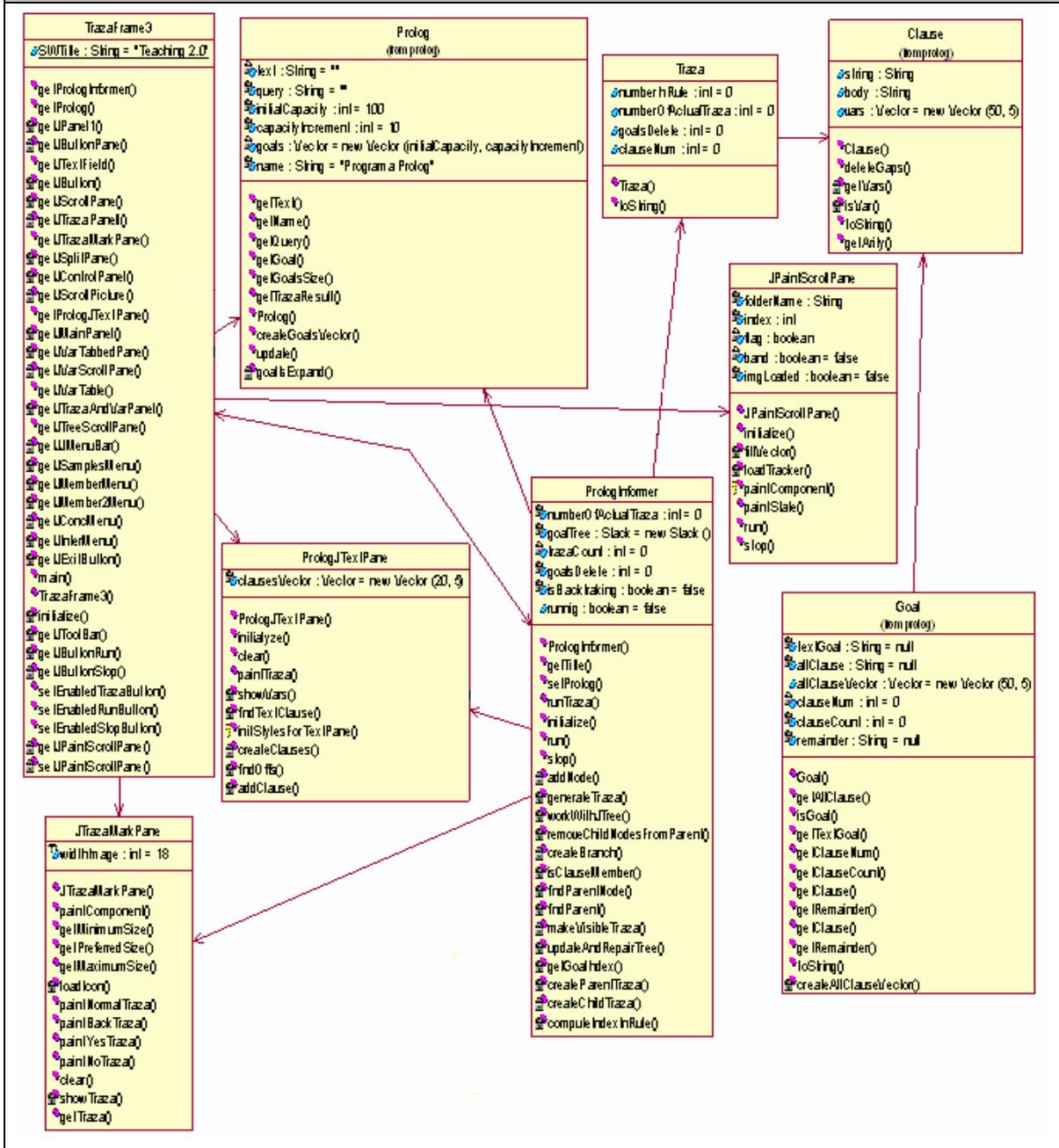


Figura. 2.5 Diagrama de clases del sistema.

A continuación se hace una breve descripción de algunas clases generales que interactúan en el sistema. Existen tres grupos de clases: Las clases que representan la

traza del “WProlog”; las clases visuales y las clases que manejan ambas.

| Clase | Propósito |
|------------------------|---|
| TrazaFrame3 | Esta clase es la que contiene todos los componentes visuales, es donde se crean y se inicializan. |
| Prolog | Esta clase es la que contiene la “query”, el resultado, y la lista de goals, como también el nombre que se va a mostrar en la barra de título cuando se esté ejecutando el ejemplo. |
| Goal | Esta clase se construye a partir de un String que comienza con “Goal:”. Tiene cláusulas que son representadas por la clase Clause , todas están almacenadas en un vector y en un String . Otro atributo es la cláusula que identifica al Goal , que es la que utiliza para tratar de unificar. Además, contiene la cantidad de cláusulas de este tipo escritas en el programa Prolog y el lugar que ocupa. |
| Clause | Al constructor de esta clase se le pasa como parámetro la cláusula en forma de String . Con la función deleteGaps(String str) se “borran” todos los espacios en blancos. El atributo string es el parámetro con el cual se construyó la cláusula pero sin espacios en blancos. En el vector vars se encuentran todas las variables de esta cláusula. Tiene una función que devuelve el “arity” de la cláusula. |
| PrologJTextPane | PrologJTextPane hereda de JTextPane . Esta clase es la que muestra el texto escrito en Prolog que se escribe con la función initialize(String t) ; esta función busca todas las cláusulas contenidas en el texto. La clase muestra las variables de la tabla y limpia los campos donde quedaron residuos. También devuelve las coordenadas donde se van a dibujar las trazas en el panel JTrazaMarkPane . |

| | |
|-------------------------|--|
| Traza | Se usa para guardar el estado por donde va la traza, apunta a la cláusula en el texto, tiene los valores que se van a comparar y podemos ver si es o no backtracking. |
| JPaintScrollPane | <p>Esta clase hereda de JPanel. Es la encargada de mostrar las imágenes animadas en el panel de la derecha de la aplicación. Tiene un arreglo v donde se van a cargar las imágenes. Al constructor se le pasa una cadena con la carpeta donde están y un entero con la cantidad de imágenes.</p> <p>La función initialize() crea un objeto de tipo MediaTracker (Sun Site Tutorial) para cargar todas las imágenes en memoria y que no se demoren mostrándose. Obtiene la dimensión del panel y llama a los métodos fillVector() y loadTracker().</p> <p>La función fillVector() es la encargada de guardar las imágenes en el arreglo y de adicionarlas al objeto tracker.</p> <p>En loadTracker() se espera porque cada imagen haya sido cargada en memoria.</p> <p>La función paintComponent(Graphics g) chequea el estado del tracker y dibuja la imagen en el centro del panel. En paintState() se manda a pintar la imagen correspondiente.</p> <p>En la función pública stop() se utiliza el método flush() de la clase Image para eliminar de la memoria el estado de cada imagen, logrando que las animaciones sean mostradas desde el principio cada vez.</p> |

| | |
|-----------------------|---|
| PrologInformer | <p>Esta clase es la que maneja y muestra la información de la traza en el PrologJTextPane (y este la tabla), la marca por donde va la traza y el árbol. Tiene cuatro funciones principales que se llaman en la clase TrazaFrame3 las cuales son initialize(), que inicializa el objeto Prolog en los componentes visuales; run(), cuando comienza la ejecución; stop(), para la ejecución y runTraza(), que va mostrando paso a paso la traza.</p> <p>El atributo goalTree es una pila que contiene todos los goal que hasta ese momento se están analizando, pueden ser: padres, ramas o hijos. O sea, aquellos goal que no tienen repercusión para el nodo actual no están aquí.</p> |
| | |

Tabla 2.5 Descripción de clases del sistema.

2.5.5 Diagramas de Interacción.

Los diagramas de interacción se utilizan para modelar el comportamiento dinámico de los sistemas. Los diagramas de interacción pueden representarse a través de los Diagramas de Colaboración y/o de los Diagramas de Secuencia. Un diagrama de secuencia es un diagrama en el que se destaca la ordenación temporal de los eventos. Los diagramas de secuencia se suelen asociar a los casos de uso, mostrando como estos se realizan a través de interacciones entre sociedades de objetos. Los mensajes se corresponden generalmente a métodos de los objetos y los objetos pueden ser creados durante la ejecución.⁵⁸

⁵⁸ Jacobson, Ivar, Booch, Grady; 2000.

Diagrama de Interacción del caso de uso Visualizar Contenido y Ejemplo

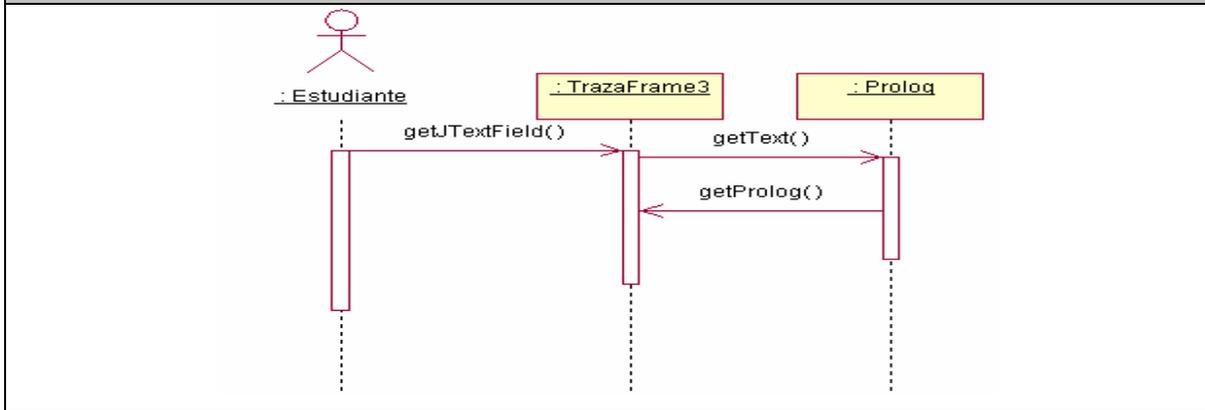


Figura. 2.6 Diagrama de Interacción Caso de Uso Visualizar Contenido y Ejemplo.

Diagrama de Interacción del caso de uso Simular Ejemplo

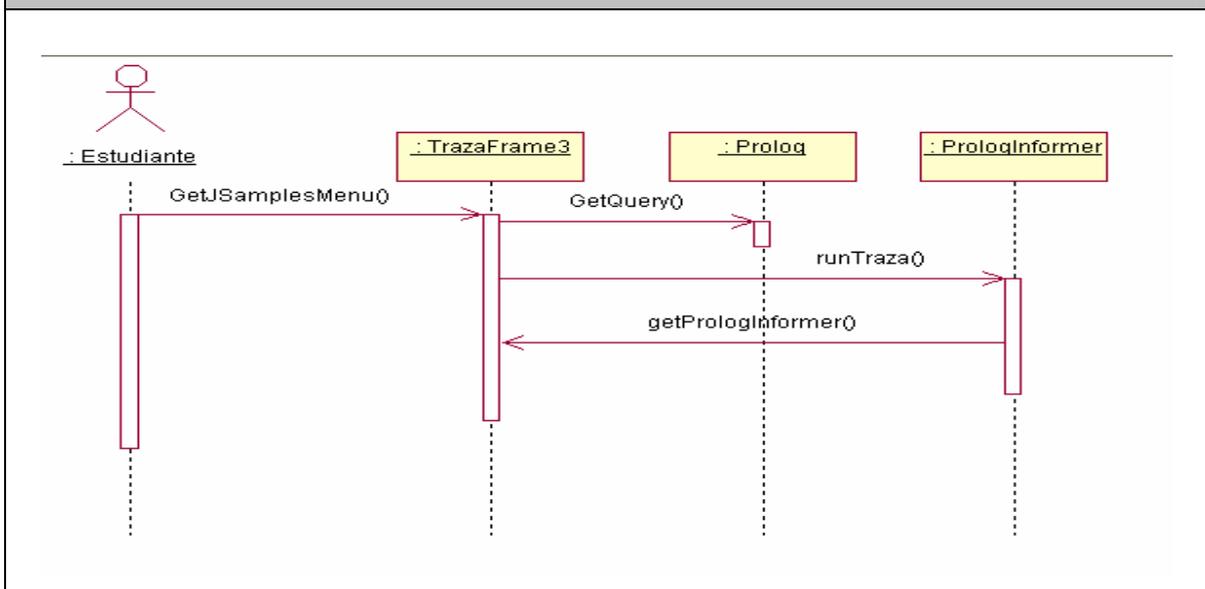


Figura. 2.7 Diagrama de Interacción Caso de Uso Simular Ejemplo.

2.6 Diagrama de Navegación

Un mapa de navegación es la representación gráfica de la organización de la información de una estructura. Expresa todas las relaciones de jerarquía y secuencia y permite elaborar escenarios de comportamiento de los usuarios.

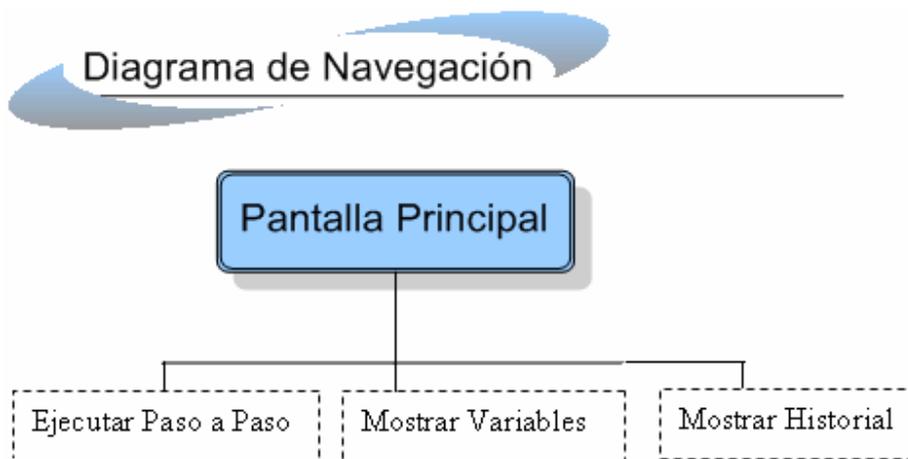


Figura. 2.8 Diagrama de Navegación.

El simulador tendrá una pantalla principal (ver anexo 5) y el menú principal de la aplicación siempre estará situado en una barra superior horizontal, para ampliar el espacio de trabajo. Desde el menú se accede a cada uno de los ejemplos, cuando se selecciona cualquiera de estos se carga la descripción del ejemplo y el cuerpo del ejemplo. En este momento se activan los botones en esta pantalla. En la pantalla principal se tiene la opción de ejecutar el ejercicio, de ver las variables e instancias en cada paso del ejemplo simulado o el árbol que se genera de las trazas; todo esto se muestra a la misma vez que se puede ver el enunciado y explicación del ejercicio (ver anexo 6).

2.7 Algunos criterios de calidad a tener en cuenta para diseñar el simulador

Cuando se hace referencia a la Calidad del Software Educativo, se requiere de un producto que satisfaga tanto las expectativas del docente como de los alumnos, a un menor costo, libre de errores y que cumpla especificaciones instruccionales y tecnológicas.

Un producto de software para ser explotado durante un largo período, necesita ser confiable, mantenible y flexible para disminuir los costos de mantenimiento y perfeccionamiento durante el tiempo de explotación.

La calidad es utilizada para caracterizar al producto, es decir, al resultado científico, el grado de solución del problema. La evaluación de la calidad de un sitio educativo permite establecer hasta que punto los componentes de la aplicación cumplen los requisitos de calidad para dar soporte al proceso de enseñanza-aprendizaje.

Los estudios realizados por Pere Márquez⁵⁹, sirven como referencia para la evaluación del simulador donde se promueve una reflexión sobre algunos aspectos funcionales, técnicos y psicológicos relacionados con los criterios de calidad para estos materiales. Dentro de los aspectos funcionales están la eficacia, facilidades de uso, múltiples enlaces. En los aspectos técnicos – estéticos se toma la calidad y cantidad de los elementos de multimedia, calidad de los contenidos, navegación, interacción, originalidad y tecnología avanzada, en cuanto a los aspectos psicológicos se asume lo atractivo y la adecuación a los destinatarios.

2.8 Valoración de la factibilidad del diseño del simulador por el criterio de expertos.

Se emplea el método de criterios de expertos como instrumento para la evaluación de la calidad y efectividad de esta investigación con el objetivo de comprobar la validez del proceso modelado teóricamente.

Se intercambiaron con profesionales, durante la realización de todas las tareas propuestas en la investigación además se tuvo en cuenta sus años de experiencias en la enseñanza superior y los resultados alcanzados en su labor profesional. Estos aportaron criterios valiosos en la constatación del problema.

A partir de ese intercambio se seleccionaron los 7 expertos que aportaron opiniones para la evaluación de la propuesta sobre su nivel de aplicabilidad, necesidad de introducción, actualidad y rigor científico. Para la selección se tuvo en cuenta la

⁵⁹ MARQUÈS G. P. Los espacios web multimedia: tipología, funciones, criterios de calidad.

experiencia en la enseñanza como profesor, en temas de aplicaciones java y programación lógica. (Ver Anexo 3).

La labor de los 7(100%) especialistas esta comprometida con el nivel de educación superior. La actividad profesional de los expertos seleccionados comprendió 2 (28.5%) profesores de que imparten la asignatura de la Programación Lógica en la UCLV, 1 (14.3%) del grupo de inteligencia artificial en la UCLV, 1(14.3%) pedagogo relacionado con la educación superior y la informática, 2 (28.5%) profesores del departamento de Ingeniería Informática en el Centro Universitario José Martí y 1 (14.3%) desarrollador de aplicaciones Java.

La guía elaborada consta de cuatro aspectos o ítems sobre los criterios, puntos de vista y argumentos a considerar por los expertos.

Como resultado del procesamiento de las respuestas a la guía, se identificaron las regularidades en relación con los juicios de mayor consenso emitidos por ellos, acerca de cada temática indagada, los que se relacionan a continuación:

El nivel de aplicación en la educación superior.

Tiene un alto nivel de aplicación porque simula el proceso de matching de la programación lógica, las cual es una asignatura que presenta dificultades en el aprendizaje y dispone de muy pocos recursos para apoyar el aprendizaje y gran parte de la bibliografía es en ingles. Brinda a los estudiantes la posibilidad de tener una herramienta disponible para comprender como funciona este proceso y leer una explicación de cada uno de los ejercicios simulados.

Necesidad de introducción.

Es una necesidad introducir este trabajo, pues en la actualidad se adolece de un espacio que de la posibilidad de simular el proceso de matching que

propicie una mejor comprensión este proceso y de la asignatura Programación Lógica. Brinda la posibilidad de ver las variables que machean una descripción de como se resuelve el ejemplo y un historial del ejemplo que se esta simulando.

Viabiliza el trabajo del estudiante pues pone en sus manos un recurso que le permite estudiar mejor esta asignatura.

La propuesta se corresponde con los enfoques actuales de educación superior cubana. Además, es exponente de un nivel científico que da respuesta a los requerimientos actuales en el campo de la investigación, por lo que puede ser aplicado.

Fiabilidad Funcional

El simulador se ejecuta en condiciones normales sin necesidad de descarga o instalación del mismo, en cualquier tipo de computadora, navegador y sistema operativo que se utiliza. Permite acceder sin dificultad a sus diferentes opciones. No es necesario instalar en la computadora del cliente la aplicación sobre la que fue elaborado, ni bajar o instalar ningún plugging.

Aspectos técnicos-estéticos

El simulador se confecciono en lenguaje Java aspecto que lo hace novedoso, esta basado en applets las cuales con muy seguras y no necesita descarga o instalación., La interfaz del software es amigable y sencilla. Todas las opciones se pueden acceder desde la pantalla principal.

Conclusiones

El estudio realizado sobre proceso de aprendizaje de la asignatura Programación Lógica propició la sustentación en la investigación realizada y el uso de un recurso informático para mejorar el aprendizaje del proceso de matching.

Se apreciaron insuficiencias en el desarrollo del aprendizaje de la programación lógica y se corroboró que el proceso de matching se considera por el educando como de una alta dificultad.

El software elaborado facilita la comprensión del proceso de matching de la programación lógica. Contribuye al aprendizaje de la programación lógica mostrando al estudiante cómo PROLOG realiza el proceso de matching en la solución de problemas; y garantiza a los usuarios estos servicios sin limitación de horario.

La validación del simulador por medio del criterio de expertos corroboró su validez para la solución del problema abordado, evidenciando que su concepción resulta en un software para el aprendizaje del alumno.

Recomendaciones

Recomienda a la dirección de la Universidad Central “Marta Abreu” de Las Villas que el simulador se publique en los servidores del centro para que se pueda acceder al mismo desde cualquier punto de la red universitaria.

Proponer a la dirección del Centro Universitario de Sancti Spíritus “José Martí Pérez” la utilización del simulador en la asignatura Programación Lógica de la carrera Ingeniería Informática.

Bibliografía

1. Alfonso, Ileana; Corral, Roberto. La tecnología educativa. El uso de las ntic en la educación. [en línea]. (accedido el 2 de marzo de 2006). Disponible cepes.uh.cu/bibliomaestria/tendenciaspedagogicas/libros/TENDE2.doc
2. Almeida S., Febles J., Bolaños O. Evolución de la enseñanza asistida por computadoras. 2005. [en línea]. (accedido el 2 de marzo de 2006). Disponible en http://www.bvs.sld.cu/revistas/ems/vol11_1_97/ems05197.htm
3. Apuntes de Java. [en línea]. (accedido el 15 de septiembre del 2006). Disponible en <http://www.monografias.com/trabajos16/java/java.shtml#browsers>
4. Arias, Leandro. La simulación computarizada como procedimiento metodológico en la enseñanza y el aprendizaje de la Electrónica. [en línea]. (accedido el 15 de marzo del 2006). Disponible en <http://www.monografias.com/trabajos13/simucos/simucos.shtml>
5. Barrera Cerna, Manuel Iván. Informática Avanzada .Aplicación de Programación Lógica en Java. [en línea]. (accedido el 6 de septiembre del 2006). Disponible en <http://html.rincondelvago.com/aplicaciones-de-prolog-en-java.html>
6. Barreiro, Enrique. El proceso de desarrollo. [en línea]. (accedido el 6 de septiembre del 2006). Disponible en <http://trevinca.ei.uvigo.es/~ebalonso/asignaturas/esx/guiones/esxClase4.pdf>
7. Bezanilla Albisua, M^a J. y Martínez Ruiz, J. A. (1996). *Bases técnico-pedagógicas para la elaboración de software educativo*. Congreso Informática Educativa 96, Universidad Nacional de Educación a Distancia.
8. Cabrera, Julio. Nuevas tecnologías, comunicación y educación. Edutec, Revista Electrónica de Tecnología Educativa, N^o 3, sep. 1996. [en línea]. (accedido el 6 de marzo del 2006). Disponible en <http://www.uib.es/depart/gte/edutece/revelec3/revelec3.html>
9. Cañas, Alberto J., Ford, Kenneth. Herramientas Para Construir y Compartir

- Modelos de Conocimiento Basados en Mapas Conceptuales. [en línea]. (accedido el 7 de septiembre del 2006). Disponible en <http://www.ihmc.us/users/acanas/Publications/RevistaInformaticaEducativa/HerramientasConsConRIE.htm>
- 10.** Casas Armengol, Miguel. (1991). La influencia de las Nuevas tecnologías de la información y la comunicación en América Latina Oportunidades del conocimiento y de la información. UNESCO. Caracas.
- 11.** Colectivo de autores. Facultad Matemática-Física-Computación. [en línea]. (accedido el 7 de octubre del 2006). Disponible en <http://www.uclv.edu.cu/uclv/Facultades/mat/?queryitem=Ciencia%20de%20la%20Computación>
- 12.** Desarrollo de aplicaciones multiplataforma con NetBeans IDE. [en línea]. (accedido el 2 de marzo de 2006). Disponible en http://www.sun.com/emrkt/innercircle/newsletter/latam/0207latam_feature.html
- 13.** Eclipse. [en línea]. (accedido el 2 de marzo de 2006). Disponible en <http://ame.endesa.es/confluence/display/AMEDev/Eclipse>
- 14.** Eclipse SDK. [en línea]. (accedido el 2 de marzo de 2006). Disponible en http://campusvirtual.unex.es/cala/epistemowikia/index.php?title=Eclipse_SDK
- 15.** Eclipse (software). [en línea]. (accedido el 2 de marzo de 2006). Disponible en [http://es.wikipedia.org/wiki/Eclipse_\(software\)](http://es.wikipedia.org/wiki/Eclipse_(software))
- 16.** El lenguaje Prolog. [en línea]. (accedido el 5 de septiembre de 2006) <http://www.etse.urv.es/EngInf/assig/iai/Laboratoris/Prolog/IntroduccionProlog.pdf>
- 17.** Escalona, Miguel. Los ordenadores en el proceso de enseñanza-aprendizaje de las ciencias. Instituto pre-vocacional de ciencias pedagógicas “Rafael Cruz Pérez”, cuba. [en línea]. (accedido el 3 de marzo de 2006). Disponible en <http://www.campus-oei.org/revista/deloslectores/997Escalona.PDF>
- 18.** Esquembre, Francisco. (2005): Easy Java Simulations. The Manual. [en línea].

- (accedido el 5 de septiembre de 2006). Disponible en http://www.um.es/fem/Ejs/Ejs_es/index.html
- 19.** Fisher, J. Prolog: Tutorial. [en línea]. (accedido el 5 de marzo de 2006). Disponible en http://www.csupomona.edu/~jrfisher/www/prolog_tutorial/contents.html
- 20.** Gálvez, Daniel. PROGRAMACIÓN LÓGICA.
- 21.** García Peñalvo, Francisco José. *HISTORIA DE JAVA*. [en línea]. (accedido el 5 de septiembre de 2006). Disponible en http://pisuerga.inf.ubu.es/lsi/Invest/Java/Tuto/I_2.htm
- 22.** García Valdivia, Zoila; Bello, Rafael. (2000) Introducción a la Inteligencia Artificial. Centro Universitario de la Ciénega.
- 23.** González, Anaisa. Modelamiento del Negocio. Centro de Estudios de Ingeniería de Sistemas (CEIS).
- 24.** 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>
- 25.** Gómez, C. El desafío de los nuevos medios de comunicación en México. AMIC, México.1992
- 26.** Gran, Manuel. La hermenéutica como esencia del proceso investigativo. Cátedra de Didáctica. 2004.
- 27.** Gutiérrez, Juan. El entorno de desarrollo eclipse. [en línea]. (accedido el 2 de marzo de 2006). Disponible en http://www.uv.es/~jgutierr/MySQL_Java/TutorialEclipse.pdf
- 28.** Hernández Sampier, Roberto. (2003). Metodología de la investigación 1. Editorial Félix Varela. La Habana.
- 29.** Hernández Sampier, Roberto. (2003). Metodología de la investigación 2. Editorial Félix Varela. La Habana.

- 30.** Herrera, René. La informatización de la sociedad: un reto para la educación cubana. [en línea]. (accedido el 1 de marzo de 2006). Disponible en <http://www.somece.org.mx/memorias/2000/docs/453.DOC>
- 31.** Introduccion a Java J2SE. [en línea]. (accedido el 11 de septiembre de 2006). Disponible en <http://diario.de/medianero.script/introduccion-a-java-j2se.html>
- 32.** Jacobson, Ivar, Booch, Grady. (2000). El proceso Unificado de desarrollo de Software. PEARSON EDUCACIÓN, S, A. Madrid
- 33.** Kofman, Hugo. Nuevas Tecnologías en la enseñanza: ¿Continentes o herramientas de la cultura humana? Universidad nacional del litoral, argentina. [en línea]. (accedido el 11 de marzo de 2006). Disponible en <http://www.rieoei.org/deloslectores/430Kofman.pdf>
- 34.** Lenguaje de programación Java. [en línea]. (accedido el 11 de septiembre de 2006). Disponible en http://es.wikipedia.org/wiki/Lenguaje_de_programaci%C3%B3n_Java
- 35.** Lino, Troadio; Alfonso, Ileana. Tendencias pedagógicas contemporáneas. [en línea]. (accedido el 2 de marzo de 2006). Disponible en <http://www.monografias.com/trabajos6/tenpe/tenpe3.shtml>
- 36.** Llorens Largo, Faraón; Castel de Haro, M^a Jesús. Prácticas de Lógica Prolog. Universidad de Alicante. [en línea]. (accedido el 5 de marzo de 2006). Disponible en <http://www.dccia.ua.es/logica/prolog/docs/prolog.pdf>
- 37.** Lógica Computacional. [en línea]. (accedido el 11 de septiembre de 2006). Disponible en <http://www.dccia.ua.es/dccia/inf/asignaturas/LC/logica/EnuncPracLC0506.pdf>
- 38.** López, Ignacio. NetBeans. [en línea]. (accedido el 11 de septiembre de 2006). Disponible en <http://ame.endesa.es/confluence/display/AMEDev/NetBeans> en <http://www.dccia.ua.es/dccia/inf/asignaturas/LC/logica/EnuncPracLC0506.pdf>
- 39.** Manzanedo del Campo, Miguel Ángel. Guía de Iniciación al Lenguaje JAVA. [en

- línea]. (accedido el 11 de septiembre de 2006). Disponible en http://pisuerga.inf.ubu.es/lsi/Invest/Java/Tuto/VI_1.htm
- 40.** Martí García, Joaquín. Puntos fuertes de Flash. [en línea]. (accedido el 11 de septiembre de 2006). Disponible en http://www.wikilearning.com/puntos_fuertes_de_flash-wkccp-69-4.htm
- 41.** Otero, Abraham. Java2, tutorial de javahispano. [en línea]. (accedido el 5 de septiembre de 2006). Disponible en <http://departamentos.unican.es/macc/asignaturas/java/Referencias/java/JavaBasico.pdf>
- 42.** Pedagogía [en línea]. (accedido el 2 de marzo de 2006). Disponible en <http://es.wikipedia.org/wiki/Pedagog%C3%ADa>
- 43.** Percepción del aprendizaje desde el adulto mayor. [en línea]. (accedido el 2 de marzo de 2006). Disponible en <http://www.monografias.com/trabajos29/aprendizaje-adulto-mayor/aprendizaje-adulto-mayor.shtml>
- 44.** Pérez Carrillo, Welkis Yazle; Gutiérrez Rodríguez Andrés Eduardo. (2006). Un método para elaborar simulaciones del lenguaje Prolog. Facultad Matemática Física y Computación. Universidad Central “Marta Abreu” de las Villas.
- 45.** Programación lógica. [en línea]. (accedido el 2 de marzo de 2006). Disponible en http://es.wikipedia.org/wiki/Programaci%C3%B3n_L%C3%B3gica
- 46.** Romo, Abel. El enfoque sociocultural del aprendizaje de Vygotsky. [en línea]. (accedido el 3 de marzo de 2006). Disponible en <http://www.monografias.com/trabajos10/enso/enso.shtml>
- 47.** Salinas Caro, Patricio. Tutorial de UML. [en línea]. (accedido el 7 de septiembre del 2006). Disponible en <http://www.dcc.uchile.cl/~psalinas/uml/introduccion>.
- 48.** Sánchez, Jaime. (1999). Construyendo y aprendiendo con el computador. Santiago: Universidad de Chile, Proyecto Enlaces.

- 49.** Spinak, Silvia; Maggioli, Susana. Estudio del mercado de software para educación interactiva. [en línea]. (accedido el 1 de marzo del 2006). Disponible en www.crnti.edu.uy/05trabajos/normas1/software.doc
- 50.** Sve, Julian. Tutorial de Flash 5. [en línea]. (accedido el 5 de septiembre de 2006). Disponible en http://www.svetlian.com/Webmaster/flash_tutor1.htm
- 51.** Torres, R.M. 2001c. La profesión docente en la era de la informática y la lucha contra la pobreza. Santiago: UNESCOOREALC. [en línea]. (accedido el 2 de marzo del 2006). Disponible en http://www.unesco.cl/medios/biblioteca/documentos/analisis_prospectivas_lac_es_p.pdf?menu=/esp/biblio/docdig
- 52.** 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>
- 53.** Unificación. [en línea]. (accedido el 2 de marzo de 2006). Disponible en http://www.monografias.com/trabajos/iartificial/pagina4_21.htm
- 54.** Vecino Alegret, Fernando. (1996). Conferencia regional sobre políticas y estrategias para la transformación de la Educación Superior en América Latina y el Caribe. Editorial Félix Varela. La Habana.
- 55.** 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

Anexos

Anexo 1

Encuesta aplicada a los estudiantes de 4to año de la Licenciatura en Ciencias de la Computación.

Estudiante:

Por estar usted recibiendo la asignatura Programación Lógica ha sido seleccionado para responder algunas preguntas que permitirán conocer los problemas que existen con el aprendizaje del proceso de matching en la Programación Lógica. La sinceridad de sus respuestas garantizará la realidad de los resultados. Está encuesta es anónima.

Objetivo: Caracterizar el aprendizaje del proceso de matching de la asignatura Programación Lógica.

| MARQUE CON UNA "X" | | | |
|---|------------------|--------------|-------|
| 1- ¿Te has interesado por el tema de la programación lógica anteriormente? a) Si ___ b) No ___ | | | |
| 2- Como consideras que es el estudio de la asignatura | | | |
| Fácil | Manejable | Difícil | |
| 3- El cubrimiento bibliográfico de las asignaturas es: | | | |
| Escaso | No muy abundante | Abundante | |
| 4- La comprensión del proceso de matching te resulta a) Fácil ___ b) Normal ___ c) Difícil ___ | | | |
| 5- Como consideras que es la orientación del estudio de la asignatura | | | |
| Buena | Regular | Insuficiente | |
| 6- Conoce usted algún recurso informático accesible para comprender mejor el proceso de matching | | | |
| Si | | No | |
| 7- Si su respuesta es afirmativa seleccione que tipo de recurso informático ha utilizado | | | |
| Tutorial Inteligente | Simulador | Entrenador | Otros |

Anexo 2

Entrevista para la obtención de los requerimientos funcionales del software.

Objetivo: Diseñar un simulador que apoye el aprendizaje del proceso de matching de la programación lógica.

Entrevistado: Msc. Lydia Rosa Ríos.

Cargo: J'Dpto Informática del CUSS

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

Pregunta1: ¿Tiene previsto algún nivel de autenticación de usuario?

Respuesta: No, el sistema puede ser accedido por todo aquel que tenga acceso al mapa conceptual donde estará insertado finalmente y no es necesario porque no se requiere seguir la traza del estudiante.

Pregunta 2: ¿Quiénes serán los usuarios finales del sistema?

Respuesta: Los usuarios finales del sistema serán los estudiantes. El software debe apoyar el proceso de aprendizaje de los mismos en la signatura de la programación lógica.

Pregunta 3: ¿Cuáles son las funciones principales que debe reunir el sistema?

Respuesta: La función principal del sistema es que debe mostrar la resolución de un ejercicio paso a paso tal y como lo hace Prolog. Además debe mostrar una explicación del mismo, con las variables y el valor que toman en cada momento.

Pregunta 4: ¿Cómo debe ser el diseño gráfico del sistema?

Respuesta: El diseño gráfico del software se debe ajustar a la apariencia del mapa conceptual donde se insertará, para ello ya se han definido algunos colores y logotipos que es necesario respetar.

Anexo 3

Entrevista a profesores que han impartido la asignatura de Prolog.

Objetivo: Caracterizar el aprendizaje del proceso de matching de la asignatura Prolog.

Preguntas:

1. ¿Cuál es su Categoría Docente o de Investigación?
2. ¿Qué tiempo lleva impartiendo la asignatura?
3. ¿Cuáles son los tipos de materiales con que el estudiante cuenta para el estudio independiente?
4. ¿Qué vías utilizan los alumnos para acceder a la bibliografía necesaria?
5. ¿Cómo es la motivación de los estudiantes por el tema de matching?
6. ¿Cómo son los resultados académicos de los estudiantes?
7. ¿Cuáles son los métodos complementarios que utiliza en la clase para apoyarse?

Anexo 4

Tabla Composición del Grupo de Especialistas consultados.

| Nombre y Apellidos | Entidad | Labor |
|------------------------------|--|---|
| Dra. Zenaida García | Universidad Central “Martha Abreu” de Las Villas Facultad Matemática Física y Computación. | Profesora de la asignatura Programación lógica. |
| Dr. Mateo Lezcano | Universidad Central “Martha Abreu” de Las Villas Facultad Matemática Física y Computación. | Grupo de Ingeniería Artificial |
| Lic. Maikel León | Universidad Central “Martha Abreu” de Las Villas Facultad Matemática Física y Computación. | Profesor de la asignatura Programación lógica |
| MSc. Aurelio Hernández Reyes | Centro Universitario “José Martí” de Sancti Spíritus | Profesor Departamento de Ingeniería Informática |
| Jorge Fardales | Centro Universitario “José Martí” de Sancti Spíritus | Profesor Departamento de Ingeniería Informática |
| Vladimir Caballero | Centro Universitario “José Martí” de Sancti Spíritus | Profesor Área de Informatización CUSS |
| Guido Cecilio | Centro Universitario “José Martí” de Sancti Spíritus | Profesor Área de Informatización CUSS |

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 y por el nivel docente metodológico que posee, para que dé sus valoraciones sobre la propuesta: “Simulador para apoyar el aprendizaje del proceso de matching”.

I. Datos Generales:

Nombres y Apellidos: _____

Centro de trabajo: _____

Años de experiencia en educación: _____ años.

Como docente: _____ años.

Como informático _____ años.

Como dirigente _____ años.

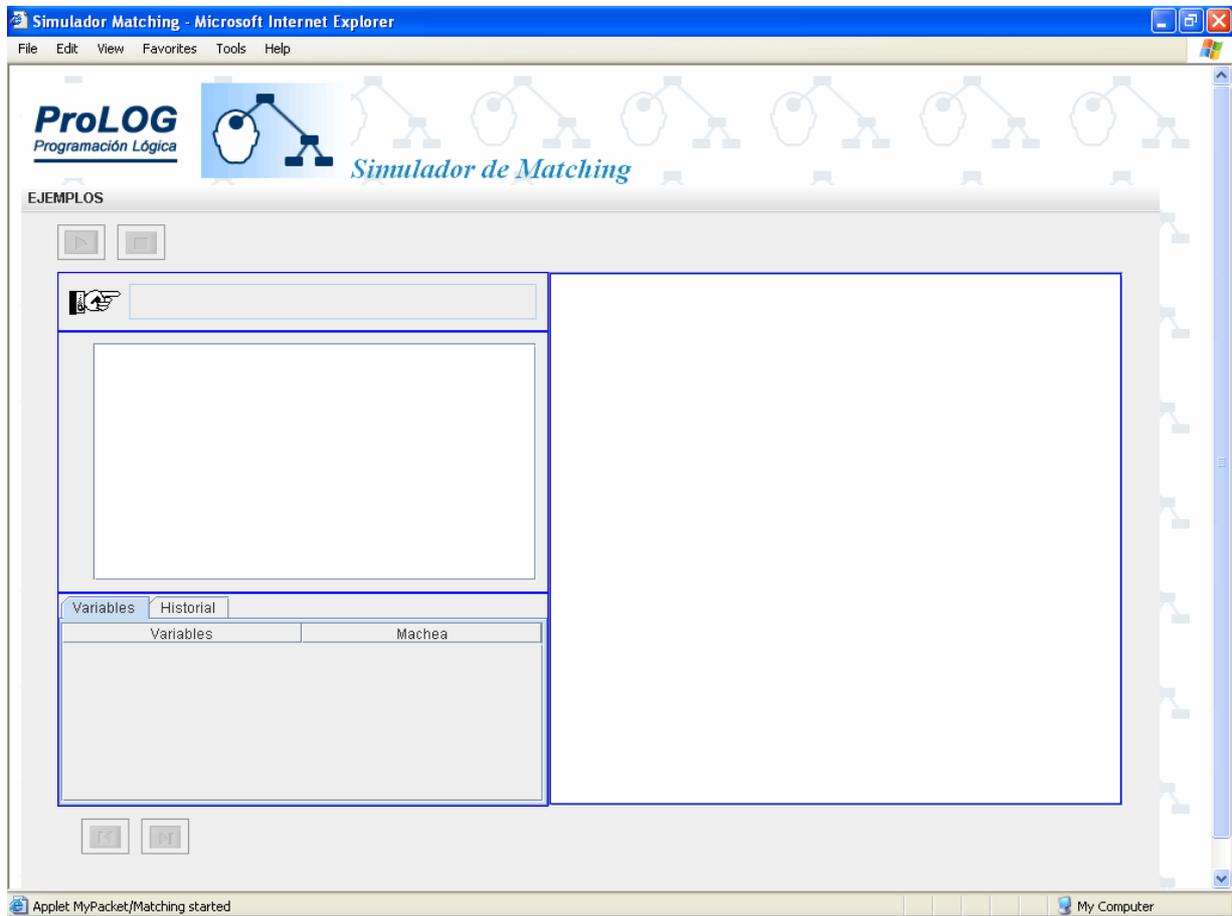
II. Sobre la propuesta usted podrá expresar su criterio teniendo en cuenta los siguientes aspectos:

1. Nivel de aplicabilidad en la práctica pedagógica.
2. Necesidad de su introducción(actualidad del tema)
3. Fiabilidad Funcional (eficiencia, facilidades de uso, múltiples enlaces)
4. Aspectos técnicos(calidad, navegación)

Gracias por su colaboración.

Anexo 6

Vista Principal de la Aplicación



Anexo 7

Vista de la simulación de un ejemplo

Applet HTML Page - Microsoft Internet Explorer

File Edit View Favorites Tools Help

ProLOG
Programación Lógica

Simulador de Matching

EJEMPLOS

abuelo(luis,martha).

```

padre(luis,carlos).
padre(luis,dalia).
padre(carlos,maria).
padre(dalia,martha).
abuelo(X,Y):- padre(X,Z),padre(Z,Y).

```

| Variables | | Historial | |
|-----------|--------|-----------|--|
| Variables | Machea | | |
| X | luis | | |
| Y | | | |
| Z | martha | | |
| | | | |
| | | | |

Y- martha Sucede lo mismo con Y y martha

**padre(luis, Z) Se pasa a probar la condición padre(X,Z), pero X esta instanciada con luis.

Luis=luis el átomo luis hace matching con el átomo luis

X-carlos la variable X hace matching y queda instanciada con el átomo carlos.

Padre(carlos,martha) No hay nada en el programa que permita llegar a conclusión, por lo que se produce un regreso atrás hasta **;

Z-dalia Con esta nueva solución se vuelve otra vez hacia delante, para para buscar. La prueba de la segunda condición de la regla expresada en la cláusula 5

Padre(dalia, martha)

Dalia=dalia

Martha =martha

Applet MyPacket/Matching started

My Computer