

**UNIVERSIDAD DE PINAR DEL RÍO**



**TÍTULO: “FUNDAMENTACIÓN DE UN SISTEMA QUE CONTRIBUYA AL  
DESARROLLO DEL PENSAMIENTO LÓGICO, EN EL MARCO  
DEL PROCESO DE ENSEÑANZA DE LA ASIGNATURA  
PROYECTOS INFORMÁTICOS.”**

**Tesis presentada en opción al Título Académico de Master en  
Nuevas Tecnologías para la Educación**

**Autor: Lic. Mayenny Linares Río**

**Tutor: Dr. Santiago Jorge Rivera**

**Pinar del Río, 2007  
“Año 49 de la Revolución”**

11 de diciembre del 2007.  
“Año 49 de la Revolución”

### **DECLARACIÓN DE AUTORÍA.**

Yo: Mayenny Linares Río, declaro que soy la única autora de este proyecto y hago constar que autorizo a la Universidad de Pinar del Río: “hermanos Saíz Montes de Oca”, ha hacer pleno uso del mismo, en lo que estime conveniente.

---

**Mayenny Linares Río**



Mayenny Linares Río  
Empresa de Materiales de Construcción de Pinar del Río  
Email: emcpr@enet.cu

## **Resumen**

En este trabajo se fundamenta la elaboración de un software educativo para el desarrollo del pensamiento lógico del alumno, a través de la implementación de un sistema de tareas en la asignatura Proyectos Informáticos. Para su cumplimiento, se pondrán en práctica varios métodos de investigación dentro de los que encontramos los métodos teóricos, los empíricos y los estadísticos.

Al dar cumplimiento a estos objetivos se espera disponer de un recurso que contenga un sistema de tareas en función de desarrollar el pensamiento lógico de nuestros estudiantes como vía para lograr una mejor calidad de graduados.



## Índice

Contenidos	Pág.
Introducción.....	1
Capítulo I. Caracterización del objeto de estudio .....	8
Conclusiones del Capítulo I.....	19
Capítulo II. Diferentes vertientes del problema y análisis de la propuesta de solución.....	20
Epígrafe 2.1. Bases teóricas para el desarrollo del pensamiento lógico.....	20
2.1.1. Pensamiento Lógico.....	20
2.1.2. Habilidades.....	28
2.1.3. Habilidades Informáticas.....	30
2.1.4. Softwares Educativos.....	34
2.1.5. Entrenador.....	39
Epígrafe 2.2. Ingeniería del software aplicada al recurso.....	44
Epígrafe 2.3. Análisis de viabilidad y costo.....	56
Conclusiones del capítulo II.....	65
Capítulo III. Diseño e implementación de la propuesta de solución.....	67
Epígrafe 3.1. Análisis comparativo de la herramienta empleada para la implementación práctica del recurso.....	67
Epígrafe 3.2. Descripción de las herramientas utilizadas para la implementación del software.....	74
Epígrafe 3.3. Implementación práctica del recurso.....	77
Conclusiones del Capítulo III.....	82
Conclusiones.....	83
Recomendaciones.....	84
Referencias bibliográficas.....	85
Anexos.	



## **Introducción**

Existe un grupo de principios que caracterizan los modelos educativos en la sociedad actual:

- ♣ Aprender a aprender
- ♣ Aprender a hacer.
- ♣ Aprender a vivir juntos.
- ♣ Aprender a ser.

Estos principios actualmente no pueden llevarse a cabo sin tener en cuenta el uso de las TIC debido al gran avance científico-tecnológico que se lleva a cabo en el mundo y a su gran impacto en la sociedad ya que ellas propician:

### **Espacio del profesor:**

- ♣ Herramientas del curso.
- ♣ Herramientas para la lección.
- ♣ Herramientas para la organización y gestión de los datos.

### **Espacio del alumno:**

- ♣ Herramientas para el trabajo asincrónico.
- ♣ Herramientas para el trabajo sincrónico.
- ♣ Herramientas de auto evaluación y seguimiento del comportamiento académico del alumno.



### **Espacio de administración:**

- ♣ Herramientas de instalación.
- ♣ Herramientas de sistema.
- ♣ Soporte de ayuda a profesores y alumnos.

### **De hecho su uso tiene muchas ventajas tales como:**

1. El centro es ubicado en el grupo que trabaja de forma cooperada en la construcción del conocimiento, el alumno desempeña un papel adaptable de acuerdo a las necesidades que van surgiendo, y la tecnología fundamental son las computadoras conectadas en redes.
2. En este modelo el factor tiempo ya no será un impedimento pues la enseñanza asincrónica, en la que el alumno accede a la información en el momento que pueda y lo desee, liberándolo del imperativo de tiempo.
3. Las relaciones tradicionalmente verticales entre profesores y alumnos van a evolucionar hacia un modelo más horizontal en el que el docente se transforma en facilitador, experto, colega y el alumno pasa a ser naturalmente activo.

Las NTIC se ponen en práctica en todo tipo de entidad o institución, principalmente en los centros educacionales y los Politécnicos de Informática no son una excepción. En este tipo de centro se lleva a cabo un proceso de enseñanza y aprendizaje a gran escala, por la cantidad de alumnos insertados en el mismo y por la preparación que en ellos se recibe.

La dinámica con que evoluciona la informática en el mundo actual, así como el carácter estratégico de la misma para el desarrollo económico-social del país, aportan los elementos que sirven de base para delimitar con mayor precisión el campo de trabajo y las funciones de los futuros egresados de esta carrera, lo cual se contempla dentro del problema definido por la Academia de Ciencias de Cuba como Perfeccionamiento del Sistema Nacional de Educación, aplicado en lo particular, al Técnico Medio en Informática.



El Técnico Medio en Informática, con una sólida preparación debe enfrentar los problemas de su profesión y resolverlos exitosamente, demostrando con ello independencia y creatividad.

Pero la realidad es otra ya que estos estudiantes no logran asimilar los contenidos, que se les imparte en las asignaturas técnicas, de forma idónea, de ahí que se aplicaran un grupo de métodos empíricos para determinar las posibles causas que propiciaban esta situación:

Al analizar el resultado de los instrumentos aplicados, determinamos como **problema científico**, el siguiente:

¿Cómo contribuir al desarrollo del pensamiento lógico de los estudiantes del Politécnico de Informática: “Carlos Hidalgo Díaz”, en el proceso de enseñanza de la asignatura Proyectos Informáticos?

De ahí que nuestro principal **objeto de estudio** sea: Proceso docente educativo de la asignatura Proyectos Informáticos, para los estudiantes de primer año del politécnico de Informática: “Carlos Hidalgo Díaz”.

#### **Campo de acción:**

Desarrollo del pensamiento lógico de los estudiantes del Politécnico de Informática: “Carlos Hidalgo Díaz”

Para poder incidir en este campo de acción nos proponemos como **Objetivo General:**

Elaborar un software educativo que contribuya al desarrollo del pensamiento lógico de los alumnos con apoyo de un sistema de tareas, en la asignatura Proyectos Informáticos.

Para el desarrollo de este proyecto es necesario el cumplimiento de los siguientes **Objetivos Específicos:**



1. Elaborar un software entrenador **PENSOLOG** que contribuya al desarrollo del pensamiento lógico de los alumnos con apoyo de un sistema de tareas, en la asignatura Proyectos Informáticos.
2. Creación de una base de datos, acumulativa de la información pertinente, para el desarrollo del software.
3. Elaborar las actividades metodológicas que incluyen las tareas de apoyo al software PENSOLOG, en la asignatura Proyectos Informáticos.

#### **Ideas a defender:**

1. El proceso docente educativo de la asignatura Proyectos Informáticos ha de tomar en cuenta el uso de un software educativo en el que se integren las habilidades profesionales en aras de contribuir al desarrollo del pensamiento lógico.
2. El software educativo a implementar en la asignatura Proyectos Informáticos debe contener un sistema de tareas que respondan a los intereses de los estudiantes, encaminados a desarrollar habilidades específicas de la asignatura, orientados a la búsqueda de vías de solución de manera lógica, y relacionados con la vida práctica.

Por lo que se propone el siguiente accionar:

#### **Tareas científicas:**

1. Constatación del problema a través de los instrumentos de investigación
  2. Analizar los modelos existentes relacionados con el tema.
  3. Estudio teórico del contenido y las habilidades desde una concepción psicológica, didáctica e informática.
  4. Fundamentar la propuesta desde los presupuestos teóricos (psicológicos, epistemológicos, didácticos- pedagógicos e informáticos).
  5. Estructuración de la propuesta metodológica que se propone.
- ♣ Determinar las habilidades que contribuyen al desarrollo del pensamiento lógico del alumno.





- ♣ Ordenar dichas habilidades por complejidad y teniendo en cuenta su carácter de sistema.
- ♣ Crear un sistema de acciones que propicie el tratamiento de dichas habilidades en clases.
- 6. Instruir a los profesores con relación a la propuesta diseñada para orientar la elaboración de un sistema de ejercicios que de salida a la misma a través de un Software educativo.
- 7. Valoración de la propuesta por los expertos.
- 8. Elaborar un Software educativo que contenga todos los elementos investigados.
  - ♣ Seleccionar el equipo de desarrollo.(Alumnos aventajados y otros con problemas de conducta para poner en práctica el aprendizaje por proyectos)
  - ♣ Planificar tareas.
  - ♣ Estimar tiempo para cada tarea.
  - ♣ Instruir, capacitar y responsabilizar con cada acción a los integrantes del grupo de desarrollo.
  - ♣ Diseño del software. (Guión)
  - ♣ Análisis de factibilidad y costo.
  - ♣ Selección de las aplicaciones con las que se desarrollará el recurso
  - ♣ Diseño gráfico y ejecución del software.
  - ♣ Control y mantenimiento.

Dichas tareas se podrán cumplir a través de la puesta en práctica de los **métodos científicos** que a continuación se mencionan.

### **1. Métodos teóricos.**

- ♣ Histórico - Lógico, para realizar el análisis del surgimiento, desarrollo y evolución del pensamiento en el hombre y la evolución histórica de la carrera de Informática, así como se ha ido desarrollando el aprendizaje de los alumnos en esta institución.
- ♣ Analítico-sintético, permitió analizar la situación actual del problema planteado para seleccionar aspectos y relacionados con esa selección incluir los que debían integrar el trabajo investigativo.
- ♣ Inductivo-deductivo, este tipo de razonamientos posibilitaron que a partir de elementos particulares relacionados con el desarrollo del pensamiento lógico se



arribara a generalidades y que su determinación permitiera arribar a particularidades del pensamiento con propósitos específicos.

- ♣ Sistémico: Con el objetivo de lograr la interacción entre los diferentes elementos que de manera ordenada conforman las acciones que tienen su nivel de salida en la estrategia que se propone.

## 2. **Métodos empíricos.**

- ♣ La observación, para constatar en la práctica la veracidad del problema planteado y la magnitud de los avances obtenidos en el transcurso de la aplicación de la estrategia.
- ♣ El cuestionario como instrumento básico de la encuesta para comprobar la opinión de profesores de elevada experiencia en la enseñanza de las asignaturas de la especialidad con propósitos específicos con relación a objetivos, contenidos, métodos, medios y la evaluación de los programas de estudio.
- ♣ Las entrevistas, del tipo grupal para comprobar las opiniones de los profesores sobre la causa que propicia el insuficiente aprendizaje por parte de los alumnos y la de los estudiantes y profesores sobre la estrategia introducida.
- ♣ La consulta a especialistas para evaluar la propuesta a que se alude.
- ♣ Análisis de documentos para extraer la información necesaria para realizar el proceso de intervención pedagógica y la posterior confección de la estrategia.

## **Novedad.**

Se ofrece un recurso para el desarrollo del pensamiento lógico del alumno, en el proceso docente educativo de la asignatura Proyectos Informáticos, en la Carrera del Técnico medio en Informática, donde se hagan partícipes a profesores y alumnos, lo que propicia un enriquecimiento teórico de la Metodología de la Enseñanza de la Informática.

## **Aportes teóricos.**

Una base teórica para la formación de las habilidades informáticas que contribuyen al desarrollo del pensamiento lógico del alumno; determinadas y clasificadas en grupos, en el



proceso docente educativo de la computación que permiten la formación de la cultura informática, en los politécnicos de esta especialidad.

### **Aportes prácticos.**

- ♣ Una propuesta metodológica para la formación de las habilidades informáticas que desarrollen el pensamiento lógico del alumno; basado en las bases teóricas propuestas.
- ♣ Un software educativo que propicie poner en práctica dicha propuesta.

### **Resultados a alcanzar.**

Elevar el desarrollo del pensamiento lógico de los estudiantes, en el proceso docente educativo de la asignatura Proyectos Informáticos, propiciando así un mejor desempeño en clases de los educandos.

### **Beneficios.**

Los alumnos aprenden más y los profesores estarán más preparados logrando mejores resultados en el aprendizaje y el centro y a su vez el país contará con una mejor calidad de graduados.

De forma general esta tesis está estructurada en tres capítulos. En un primer capítulo se aborda la caracterización del objeto, o sea la evolución histórica de la institución y de la asignatura, así como la situación actual de los programas, la preparación de los profesores para enfrentar la problemática planteada y el aprendizaje de los alumnos; en el capítulo dos se plasman las bases teóricas de la problemática descrita, así como el diseño teórico para su solución y el análisis de viabilidad y costo, y en el tercer capítulo se muestra el análisis comparativo de la herramienta utilizada para la implementación del recurso descrito, su implementación práctica y la descripción de otras aplicaciones utilizadas para su elaboración.



## **Capítulo I. Fundamentación del problema.**

### **1.1. Antecedentes históricos de la carrera. (3)**

En el año 1978 por Resolución Ministerial 316 del 7 de julio de dicho año se aprueba el plan de estudio de la especialidad Programación para Computadoras para aplicarse en 3 centros de las provincias Ciudad de la Habana, Villa Clara y Santiago de Cuba a partir de alumnos con nivel de ingreso 12 grados. Este plan de estudio con algunas variantes estuvo vigente, hasta el curso 1998 – 1999.

En la Resolución Económica del V Congreso del Partido Comunista de Cuba, de octubre de 1997, queda declarada la importancia actual de la informática y la industria del software al expresarse:

*“El país debe encaminarse resueltamente a la modernización informática mediante un programa integral que involucre a las organizaciones que deben proveer los recursos materiales, financieros e intelectuales y a las entidades económicas, políticas y sociales que deben traducirlos en más y mejores productos y servicios.”*

En el curso 1999- 2000 se moderniza la Base Material de estudio y se imparten cursos de actualización a los 59 profesores de la especialidad en los 8 centros existentes en el país, además se dictan las Resoluciones Ministeriales 48 y 50 del 14 de abril del 2000 para dar respuesta a las necesidades de formación de técnicos en informática.

La RM 50 modifica la especialidad Programación para Computadoras convirtiéndola en la Especialidad Informática con un perfil que le permitía al graduado participar como programador en equipos de producción de software, administrar pequeñas redes y operar eficientemente microcomputadoras. Este plan de estudio contaba con un total de 4 800 horas, de ellas, 560 dedicadas a asignaturas de Formación General, 2160 a asignaturas Técnicas y 2080 a prácticas de las cuales 1600 se realizaban en las empresas.



La RM 48 fue creada, al no contarse con suficiente cantera de alumnos de 12 grado y permite el ingreso con 9no grado aprobado, para formar un operador avanzado de microcomputadora. Este plan de estudio contaba con un total de 6360 horas, de ellas, 1880 dedicadas a asignaturas de Formación General, 2400 a asignaturas Técnicas y 2400 a prácticas de las cuales 800 se realizaban en las empresas.

El plan comienza a aplicarse en septiembre del 2000, lo cual permitió extender la especialidad a todas las provincias con una matrícula de 3056 alumnos y se utilizaron como profesores, a licenciados de otras especialidades que impartían la asignatura Computación en los centros politécnicos, los cuales fueron reorientados hacia la informática mediante un curso recibido al efecto, además de alumnos de la especialidad Programación para Computadoras que fueron habilitados para asumir la tarea.

El 23 de febrero del 2001 a solicitud de la dirección del país se dicta la Resolución Ministerial 18 para formar un técnico con el perfil aprobado en la RM 50 con nivel de ingreso 9no grado, que fueran previamente seleccionados. Este plan de estudio contaba con un total de 6480 horas, de ellas, 1880 dedicadas a asignaturas de Formación General, 2560 a asignaturas Técnicas y 2040 a prácticas de las cuales 800 se realizaban en las empresas. El plan se aplica de inmediato a los alumnos que transitaban por la RM 48 del 2000 y en septiembre del 2001 aumenta la matrícula a 11363 alumnos en 36 centros del país. En este curso comienza la formación emergentes de profesores de la especialidad con 576 compañeros de ellos 453 alumnos graduados o cursando las prácticas pre-profesionales de diferentes carreras de Técnico Medio y se crea la carrera de Licenciatura en Educación, Especialidad Informática, en la modalidad de cursos para trabajadores y se formaron colectivos de autores para elaborar 5 libros de textos para la especialidad.

En el curso 2002-2003, a solicitud del MIC, se modifica el plan de estudios de la especialidad extendiendo las prácticas pre-profesionales para los alumnos con nivel de ingreso 9no grado de seis meses a un año surgiendo la RM 129 del 22 de julio del 2002. Este plan de estudio cuenta con un total de 5800 horas, de ellas 1720 dedicadas a asignaturas de Formación



General, 1860 a asignaturas Técnicas y 2320 a prácticas de las cuales 1600 se realizan en las empresas lo que constituye una alternativa ante la falta de equipos y profesionales en los centros. El plan se aplica de inmediato a los alumnos que transitaban por la RM 18 del 2001 y al nuevo ingreso de septiembre del 2002 elevándose la matrícula a 21620 alumnos y los profesores a 1146 utilizando las mismas vías de formación. El número de centro creció a 47. En este propio curso se aprueba la Carta Circular 8 del 2002 estableciendo la continuidad de la preparación de los profesores licenciados para que en un año y medio se reorienten a profesores de la especialidad Informática lo que se cumplió solo en parte por falta de profesionales en los pedagógicos.

En el curso 2003-2004 se eleva la matrícula a 30030 con la siguiente estructura por años: 1er año, 9214; 2do año, 10 901; 3er año, 8299; 4to año, 1616. En este año se produce la primera graduación de alumnos con nivel de ingreso de 9no grado. El número de profesores se elevó a 1665, la gran mayoría emergentes y a 52 los centros de informática.

En este curso escolar se aprueba por la Resolución Ministerial 129 del 1ro de septiembre del 2004, a fin de introducir las transformaciones de la Educación Técnica y Profesional, un nuevo plan de estudio parcial para aplicar a los alumnos que ingresaron en el curso 2004-2005 para que reciban todos los contenidos que se imparten en el Pre de Ciencias, lo que conlleva a una reducción de las horas de asignaturas técnicas. En este curso la matrícula se elevó a 38 220 alumnos y se proyecta una graduación de más de 8000 alumnos, el número de centros aumentó a 55.

**Como puede apreciarse, el incremento del número de alumnos que ha arribado a los años superiores de la carrera, la falta de cobertura en equipos, de profesores bien preparados y de conectividad no ha permitido el cumplimiento satisfactorio de los objetivos trazados en los planes y programas de estudios.**



Teniendo en cuenta lo anterior, el plan de estudio actual, contribuye a la formación de los recursos humanos profesionales necesarios para dar repuesta a las necesidades que plantea el desarrollo político, económico y social del país aplicando las experiencias y métodos de trabajo de la UCI, dentro de los que encontramos el aprendizaje por proyectos.

## **1.2. Antecedentes históricos de la disciplina proyectos Informáticos.**

Los elementos que componen esta disciplina estuvieron presentes desde el primer plan de estudio elaborado para la especialidad de Informática en la enseñanza politécnica media superior, dada su importancia en la formación del graduado de este perfil. La participación de los estudiantes en la realización de proyectos informáticos se concentraba, como norma en el último curso, mediante la realización de las prácticas pre-profesionales, en empresas de la producción y los servicios. Estas prácticas, casi siempre, concluían con la presentación y defensa de un software o parte de una aplicación informática.

Experiencias puntuales demuestran que es posible convertir a los Institutos Politécnicos en centros de elaboración de software.

El Técnico en Informática requiere una fuerte preparación en las técnicas modernas de programación, ya que las mismas contribuyen, de manera importante, a que este egresado haga un uso eficiente y verdaderamente profesional de las computadoras. Está concebida como una disciplina central de la especialidad.

Pero si hacemos un análisis del programa de esta asignatura y de las que tributan a ella, nos encontramos con:

## **1.3. Valoración crítica de los programas.**

Al realizar un estudio del diseño curricular de la carrera de Técnico Medio en Informática, encontramos que cumple con los requisitos establecidos pues contiene:

- ♣ Antecedentes históricos.
- ♣ Fundamentación de los cambios del plan de estudio.



- ♣ Objeto de trabajo y campo de acción del técnico medio en informática

Por qué y para qué enseñar

- ♣ Plan de estudio aprobado por la resolución ministerial no. 129 del 1ro de septiembre del 2004



Cuándo y cómo enseñar – aprender

- ♣ Objetivos generales:
- ♣ Contenidos para cumplimentar los objetivos generales

Qué enseñar – aprender

- ♣ Programa de disciplina  
Especialidad informática.  
Disciplina sistemas digitales

Años en que se imparte 1ro, 2do, 3ro y 4to

1. Distribución del tiempo  
total: 966 horas  
clases: 966 horas  
componente laboral: 0 horas
2. Fundamentos de la disciplina
3. Objetivos de la disciplina
4. Asignaturas que componen la disciplina y distribución del tiempo
5. Contenido de la disciplina





6. Sistema de conocimiento.

7. Sistema de habilidades

Qué enseñar – aprender

8. Orientaciones metodológicas



Qué, cuándo y cómo evaluar.

En su estructura presenta

- ♣ Organización de los contenidos en materias o módulos.
- ♣ Estructuración de las asignaturas o módulos en un plan curricular:
  - a. Secuencia horizontal.
  - b. Secuencia vertical.
- ♣ Establecimiento del mapa curricular.
  - a. Lineal por asignaturas.
  - b. Modular.
  - c. Mixto

En la organización para la puesta en práctica (Toma como base las características de los alumnos)

- ♣ Preparación de los sujetos: Los directivos se prepararon en el mes de agosto en la UCI para enfrentar el nuevo plan y estos a su vez prepararon a los profesores.
- ♣ Comprensión de la concepción: Se organizaron reuniones con los profesores en las que estuvieron presentes directivos provinciales y nacionales, explicando el nuevo modelo y concientizando a los trabajadores con el cambio.



..... *Fundamentación del problema.*

- ♣ Creación de condiciones: El centro está inmerso en un proceso de reparación total y se le asignó un grupo de materiales (Computadoras, videos, TV, L/T, Software y otros periféricos, neveras, colchones, etcétera) necesarios para la puesta en práctica del plan.
- ♣ Preparación individual y colectiva del personal pedagógico: Se asesoró al personal docente en el mes de Julio por parte del ISP y se les fue asignado un profesor, a cada departamento, de la Universidad, encargándose este de la preparación de los profesores.
- ♣ Elaboración de horarios, grupos etc.: Se confeccionaron los grupos de no más de 30 alumnos, se les asignaron los profesores de cada asignatura y se elaboró el horario docente para cada año.

De forma general es un plan flexible pues puede sufrir cambios en dependencia de las necesidades de la escuela, diseñado en función de las necesidades del estudiante y de los intereses de la sociedad con el egresado.

Cada directivo envía de forma mensual los resultados del aprendizaje y de la organización del proceso de enseñanza y aprendizaje a los responsables de dar seguimiento al cumplimiento del diseño en el Ministerio de Educación.

Existen elementos en el currículo del cual se habla, que no satisfacen condiciones necesarias que deben aparecer como por ejemplo:

- ♣ No se tienen en cuenta las motivaciones de los estudiantes para definir los objetivos a cumplir en los programas.
- ♣ Las disciplinas de Diseño y Programación, a pesar de tributar ambas a la confección de los Software a producir por estudiantes y profesores, se tratan de forma divorciada.
- ♣ Se implementan en un mismo curso escolar más de un lenguaje de programación, lo que posibilita que el estudiante no profundice en ninguno, o sea, sabrá un poco de todo, pero no dominará un lenguaje determinado.
- ♣ A pesar de que los estudiantes comienzan con sus proyectos en primer año, ellos reciben la asignatura diseño a partir del segundo y es la asignatura de Proyectos Informáticos la que debe asumir esta responsabilidad de forma muy ligera.



- ♣ Se proponen como habilidades un grupo grande de operaciones.
- ♣ Existe un número inmenso de habilidades a desarrollar en el programa de cada disciplina, aspecto este que se hace imposible durante un curso escolar y es por esto que a veces se dice tenemos los objetivos tratados pero no vencidos.

Esta situación incide directamente en la correcta dirección del proceso docente educativo por parte del profesor.

En los politécnicos de Informática del país existe un gran número de profesores y de personal de apoyo a la docencia, de los cuales depende la preparación de los egresados en dichas instituciones.

En el Politécnico que hemos tomado como modelo contamos con una plantilla de 191 trabajadores, de ellos solo 45 forman parte del claustro de profesores de las asignaturas técnicas de primer año

**Características de los profesores de asignaturas técnicas** (ver representación en Anexo VIII)

Total Prof.	Grad. Univ.		Técnicos medios			Esc. Form. Maest.	Esc. Sup. Quím.
	Espec.	Otras	Espec.	Elect.	otras		
45	5	9	4	20	7	1	1

Con el proceso de Universalización que se lleva a cabo en nuestro país, la mayoría de ellos son profesores en formación, de ahí que se preste gran atención a su preparación general.

**La cantidad de actividades realizadas no son suficientes para lograr la preparación, con calidad, de los docentes en cuanto a las exigencias de la clase y los resultados en el aprendizaje, por lo que existen dificultades tales como:**



- ♣ No todas las actividades metodológicas se desarrollan con calidad.
- ♣ No son suficientes el número de actividades metodológicas que se desarrollan por la estructura de dirección.
- ♣ Insuficiencias en la organización y dirección del aprendizaje al no realizar una correcta planificación, orientación y control de las tareas docentes teniendo en cuenta los niveles de desempeño de los alumnos.
- ♣ No se sistematiza el trabajo con la revisión de libretas para garantizar la orientación y control de los planes correctivos, dirigido a los elementos del conocimiento con mayor dificultad.
- ♣ Insuficiencias en el tratamiento de la evaluación para que esta sirva como elemento estimulador y no desestimulador a la dirección y resultados del aprendizaje en los alumnos.
- ♣ Insuficiencias en la actualización de los documentos oficiales del maestro.
- ♣ Insuficiencias en el aprovechamiento de las potencialidades que brinda la clase para el tratamiento efectivo al TPI partiendo de los programas priorizados y directores de carácter nacional.
- ♣ No se realiza un seguimiento efectivo y sistemático a la preparación de los profesores en formación.
- ♣ Actualmente muchos de los profesores en formación, debido a problemas internos de la escuela, se desempeñan como PGI, aún los que cursan el primer año en su carrera, lo que impide su correcto desempeño en la docencia.

Este grupo de dificultades afecta el aprendizaje de los alumnos y más aún si tenemos en cuenta la gran cantidad y variedad de los mismos, de ahí que sea de vital importancia estructurar una metodología que propicie la preparación indispensable para enfrentarse a la problemática planteada. (Ver Anexo IX)

Actualmente existe una cantidad de 2450 estudiantes, con un nivel de aprendizaje alto, por pertenecer al grupo de aspirantes a IPVCE y por haber alcanzado una nota elevada en las pruebas de ingreso.



Haciendo un análisis más detallado de sus características encontramos que sus principales problemas en la familia y la comunidad están dados por el alcoholismo, y la falta de comunicación. En la escuela los principales factores de riesgos están dados por la falta de visita al hogar y su tendencia al rock, queriendo tener en el centro la apariencia de los mismos.

Este grupo de factores incide directamente en el aprendizaje de los alumnos, y más aún en los de primer año, pues se observa diferencias en el índice académico, de los mismos, en cuanto al que tenían cuando entraron al centro y al que conservan actualmente.

En la asignatura Proyectos informáticos, no mantienen calificaciones adecuadas, pues no logran adaptarse al sistema de evaluaciones, adoptado para este tipo de institución, y no se extrapolan a la realidad de las problemáticas que se les plantea en clases.

Cada una de estas dificultades se constató a través de la aplicación de los métodos empíricos.

Los instrumentos aplicados a profesores (Anexo I y II) reflejan la poca motivación de los estudiantes, las disciplinas de Diseño y Programación, a pesar de tributar ambas a la confección de los Software a producir por estudiantes y profesores, se tratan de forma divorciada, todos no conocen completamente el perfil del egresado, ni la estructura interna de

las habilidades que deben desarrollar en sus alumnos, se implementan en un mismo curso escolar más de un lenguaje de programación, lo que posibilita que el estudiante no profundice en ninguno, o sea, sabrá un poco de todo, pero no dominará un lenguaje determinado, existe un número inmenso de habilidades a desarrollar en el programa de cada disciplina, aspecto este que se hace imposible durante un curso escolar, poco desarrollo del pensamiento lógico, poco tiempo para la preparación, falta de medio de enseñanza, no existe dominio de cómo evaluar según los tres niveles de aprendizaje.



Los Instrumentos aplicados a los alumnos (Anexo III y IV), muestran que no se tienen en cuenta las motivaciones de los estudiantes para definir los objetivos a cumplir en los programas, pues tal parece que los quieren formar como matemáticos y no como informáticos.

A través de la observación a clases (Anexo V) se constata la pobre motivación por parte de los alumnos y su poco desarrollo del pensamiento lógico, así como la insuficiente apropiación del conocimiento en clases.

La revisión de documentos (Anexo VI y VII), muestra que:

- ♣ No se tienen en cuenta las motivaciones de los estudiantes para definir los objetivos a cumplir en los programas.
- ♣ Las disciplinas de Diseño y Programación, a pesar de tributar ambas a la confección de los Software a producir por estudiantes y profesores, se tratan de forma divorciada.
- ♣ A pesar de que los estudiantes comienzan con sus proyectos en primer año, ellos reciben la asignatura diseño a partir del segundo y es la asignatura de Proyectos Informáticos la que debe asumir esta responsabilidad de forma muy ligera.
- ♣ Se implementan en un mismo curso escolar más de un lenguaje de programación, lo que posibilita que el estudiante no profundice en ninguno, o sea, sabrá un poco de todo, pero no dominará un lenguaje determinado.
- ♣ Existe un número inmenso de habilidades a desarrollar en el programa de cada disciplina, aspecto este que se hace imposible durante un curso escolar y es por esto que a veces se dice tenemos los objetivos tratados pero no vencidos.
- ♣ Los resultados del aprendizaje no tienen la calidad requerida.



## **Conclusiones del capítulo I**

La asignatura proyectos Informáticos tiene gran importancia para el desarrollo de los estudiantes como futuro profesional debido a su aplicación en la solución de los problemas que se les pueda presentar en las empresas.

Existe una gran cantidad de profesores laborando en la institución, de ellos el 68,89 % son profesores en formación, el 20 % de graduados en otras especialidades, el 4,44 % que no son graduados universitarios y solo el 11,11 % son graduados universitarios en la especialidad. Esta situación trae como consecuencia que los profesores no cuenten con la preparación adecuada para enfrentarse a ciertos tipos de situaciones que se presentan en la institución, por lo que hay que hacer gran hincapié en su preparación y dar seguimiento a la misma.

Los alumnos necesitan estar bien preparados para enfrentarse a su vida profesional y para ello ha de tenerse en cuenta sus motivaciones, intereses y su correcta orientación basada en programas concebidos teniendo en cuenta los términos de diseño curricular y las fases para su elaboración.

Todo ello ratifica la necesidad de creación de un elemento de apoyo a las actividades docentes, para una mejor preparación de los alumnos en la asignatura Proyectos Informáticos, y que en este caso planteamos sea nuestro software apoyado en un sistema de tareas.



## **Capítulo II. Bases teóricas del problema.**

### **Epígrafe 2.1. Bases teóricas para el desarrollo del pensamiento lógico.**

#### **1. El pensamiento lógico.**

##### **1.1. Bases Morfológicas de la localización de las funciones en la corteza cerebral (centros corticales).**

El conocimiento de la localización de las funciones en la corteza cerebral tiene gran significado teórico, puesto que da una idea de la regulación nerviosa de todos los procesos del organismo y su adaptación al medio ambiente, así mismo tiene gran importancia práctica para el diagnóstico de los lugares lesionados en los hemisferios del cerebro.

Según Pavlov, el centro es el extremo cerebral del llamado analizador. El analizador es un mecanismo nervioso cuya función consiste en efectuar el análisis, la síntesis, la combinación de analizadores entre sí y las diferentes actividades del organismo. (Ver representación en Anexo IX)

- ♣ Analizador acústico: al medio del giro temporal superior.
- ♣ Analizador óptico: lóbulo occipital.
- ♣ Analizador olfatorio: límites de la base del rinencéfalo.
- ♣ Analizador gustativo: en el inferior del giro postcentral.
- ♣ Analizador tegumento: giro postcentral en la región parietal superior.
- ♣ Sensibilidad cutánea: corteza del lobulillo parietal superior.

El segundo sistema de señalización es el pensamiento humano, que es siempre discursivo, puesto que la lengua es la envoltura material del pensamiento. El analizador motor de la articulación del lenguaje se encuentra en la parte posterior del giro frontal inferior. (9)

#### **2. La concepción como unidad de lo sensorial y lo racional.**

La Filosofía distinguió hace ya mucho, dos elementos que constituyen la cognición: lo sensorial (las sensaciones, percepciones y representaciones) y lo racional (el pensamiento en sus diversas formas: conceptos, juicios, deducciones, hipótesis y teorías). (4)





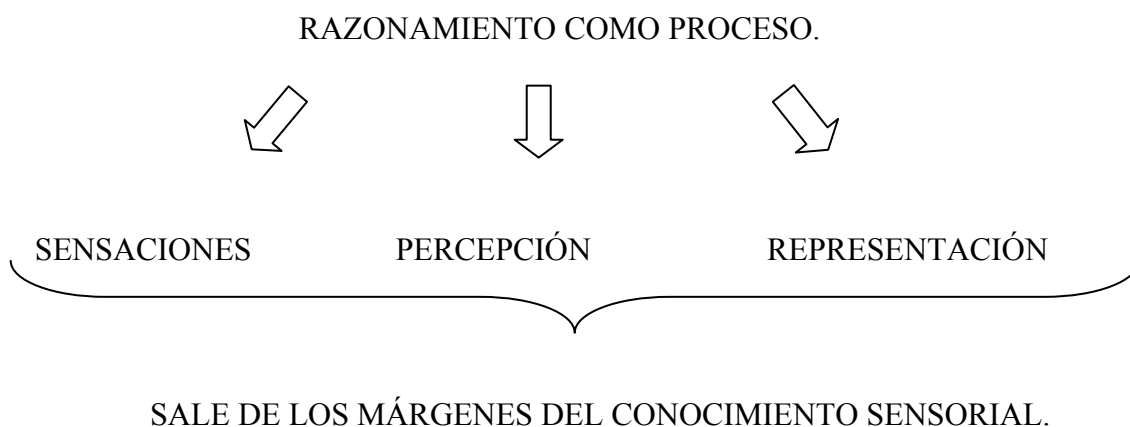
En la cognición corresponde el papel fundamental al elemento sensorial, pues el hombre está unido a través de los órganos de los sentidos.

Es sabido que las sensaciones son resultado de la interacción de dos sistemas materiales: el objeto (estímulo) que se encuentra fuera de los órganos de los sentidos y el sujeto (órganos de los sentidos, sistema nervioso), que ejerce un influjo sustancial en la forma de las sensaciones.

El apriorismo de Kant es una forma un tanto atenuada y debilitada del racionalismo. A su juicio el conocimiento tiene dos orígenes independientes entre sí: primero los datos de las percepciones sensoriales, que constituyen el contenido del conocimiento, y segundo las formas de la sensibilidad y el raciocinio que tienen un carácter apriorístico (independiente de la experiencia).

El conocimiento es la unidad del reflejo sensorial y racional de la realidad. El hombre no tiene un conocimiento real fuera de la representación sensorial.

### **El conocimiento sensorial y el pensamiento.**





#### **Realización práctica de los conocimientos. (4)**

Para enriquecer el mundo del hombre, las ideas deben materializarse, adquirir la forma de objetos o cosas. (Ver relación en Anexo XVIII)

#### **Conocimiento y valor. (4)**

Las ideas se realizan prácticamente en la cultura material y espiritual. De ahí su relación con las necesidades humanas.

La historia del pensamiento cubano muestra la existencia de la continuidad de un proceso en el cual todos los valores esenciales del hombre en función del progreso se han conservado hasta afirmarse como tradición. Tradición que garantiza la existencia y permanencia de la memoria histórica, así como su enriquecimiento, vitalidad y defensa de la identidad nacional un tanto tal. (2. Pág. 618)

Valor: propiedad de los objetos sumergidos en el proceso de desarrollo de la sociedad, y al mismo tiempo una propiedad de los objetos de la naturaleza incluidos en el proceso del trabajo y el ser. (1. pág. 259)

### **3. El pensamiento según la Psicología.**

#### **Características generales del pensamiento. (7. Pág. 292.)**

El pensamiento es el proceso psíquico socialmente condicionado de búsquedas y descubrimientos de lo esencialmente nuevo y está indisolublemente ligado al lenguaje. El pensamiento surge del conocimiento sensorial sobre la base de la actividad práctica y lo excede ampliamente.

Cuando damos propiedades a organismos o cosas que no podemos ver a simple vista estamos en presencia del pensamiento indirecto.

El pensamiento empieza allí donde el conocimiento sensorial se hace insuficiente. El pensamiento continúa desarrollando el trabajo cognitivo de las sensaciones, percepciones y representaciones, superando ampliamente sus límites.



### La determinación del pensamiento como proceso. (1. Pág. 308)

En el proceso del pensamiento se lleva a cabo el principio de selección que está dado por el recuerdo de teoremas, conceptos, definiciones y se eligen las que permiten la solución del problema.

Si se da la solución del problema se inhibirá el desarrollo de la actividad racional.

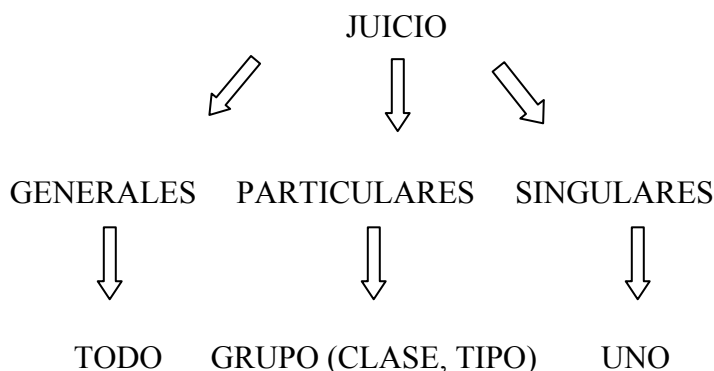
### El pensamiento y el lenguaje.

El pensamiento humano, sea cual fuere la forma en que se realice, es imposible sin el idioma. Cada pensamiento surge y se desarrolla en conexión indivisible con el lenguaje. Mientras más profunda y con mayor fundamento hay sido pensada una idea determinada, más precisa y concreta se expresa en la palabra, en el lenguaje oral y escrito.

Cuando una persona formula sus reflexiones en voz alta para los otros, la formula, además, para sí mismo. Gracias a esto se hace posible el razonamiento amplio, consecuente y sistemático, o sea, la confrontación clara y correcta de todas las ideas fundamentales que surgen en el proceso del pensamiento. La palabra, la formulación de la idea contiene de esta manera, las condiciones fundamentales más importantes del pensamiento discursivo (del latín discursus – razonamiento), o sea, del pensamiento lógico, separado y consciente basado en el razonamiento.

### Lógica y psicología del pensamiento.

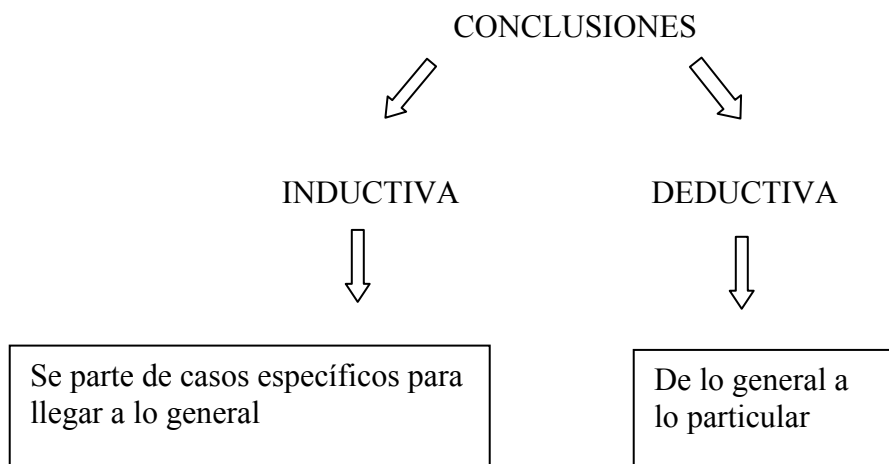
Juicio: es el reflejo de las conexiones existentes entre los objetos y fenómenos de la realidad o entre sus propiedades y características.





### Métodos básicos para formar juicios.

1. Directo: cuando se expresa lo que se percibe.
2. A través de conclusiones y razonamientos.



La lógica formal estudia las ideas, conceptos, juicios y establece relaciones entre ellos.

### Operaciones racionales fundamentales. (9)

- a) Análisis – Síntesis.

Análisis: es la identificación en el objeto de aspectos, elementos, propiedades, conexiones, relaciones, etc., es la división del objeto de conocimiento en diversas partes y componentes.

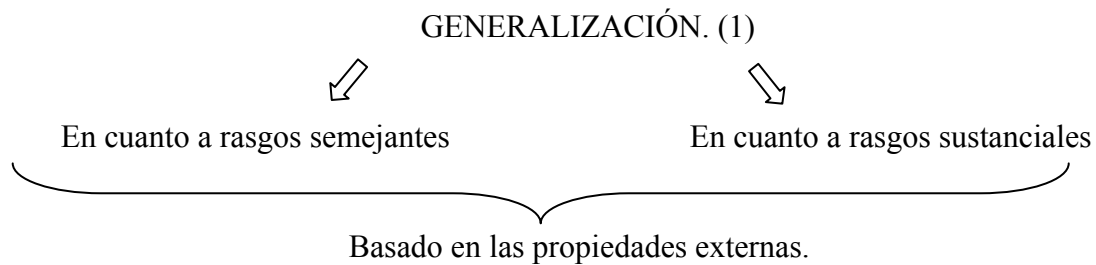
Síntesis: en este proceso se produce la unión, la correlación de los elementos en que habría sido dividido el objeto de conocimiento.

- b) Comparación: establecer semejanzas y diferencia.

Para esto es necesario separar aspectos determinados de ellos. El análisis es una de las partes constituyentes de la comparación.

- c) Generalización – Sistematización.

Generalización: separación mental de lo general en los objetos y fenómenos de la realidad, basándose en ella, en su unificación mental.



Sistematización: o clasificación de los objetos y fenómenos en su distribución mental en grupos y subgrupos, según las semejanzas y las diferencias que hay entre ellos.

d) Abstracción – Concreción.

Abstracción: Se piensa en algo por medio de la palabra haciendo caso omiso a la imagen total del objeto. Pensamiento superior específicamente humano y superior (I. V. Pavlov. Obras completas. Edición Rusa. 1951. T III. L II. Pág. 232)

Concreción: proceso opuesto de la abstracción. Se concreta sobre lo particular que corresponde a lo general determinado.

### **Motivación del pensamiento. (7)**

#### **Motivos:**

1. Específicos – Cognitivos: en él intervienen las fuerzas motrices para la actividad racional (Ej.: curiosidad) y los estímulos.
2. Acción de causas exteriores: Ej.: por obligación, por no quedarse atrás con respecto al grupo.

### **El pensamiento y la solución de problemas. (1)**

El alumno en el transcurso del tiempo se enfrenta a problemas y a situaciones problemáticas diversas, pero a veces se tiende a confundir estos términos. La situación problemática es cuando durante la actividad, la persona se enfrenta, de forma inesperada, a algo



incomprensible, desconocido, etc. Y el problema es cuando se identifica lo dado (conocido) y lo desconocido (incógnita).

### **Tipos de pensamiento. (7)**

1. Pensamiento efectivo: inicio, preescolar

Lo primario es precisamente la actividad práctica y no la actividad teórica pura.

El niño analiza y sintetiza los objetos de conocimiento a medida que los va clasificando, divide y nuevamente uniendo con las manos, vincula un objeto con otro durante la percepción de éstos que se produce en un momento dado. (Ej.: rompo para ver lo que tiene dentro)

2. Pensamiento figurativo: comienza de los 4 a los 7 años.

Falta de conceptos. En muchos casos se hace innecesaria la manipulación sistemática práctica del objeto, pero en todos los casos es necesario percibir con claridad y representar evidentemente el objeto. (Ej.: manipula la plastilina y crea un objeto que no ha tocado, solo lo ha visto)

3. Pensamiento abstracto: se expresa no solo bajo la forma de las acciones prácticas y de imágenes, sino, ante todo, en forma de conceptos y razonamientos abstractos.

### **Particularidades individuales del pensamiento.**

1. Independencia: carácter creador.
2. Flexibilidad: capacidad de cambiar el plan para resolver la tarea.
3. Rapidez: toma de decisiones en el tiempo.

La característica más importante del pensamiento es la capacidad de identificar lo esencial, de llegar independientemente a nuevas generalizaciones.

Después de analizar todo lo antes abordado podemos llegar a la conclusión de que

El pensamiento lógico es el proceso que pone en práctica una secuencia de acciones organizadas según su complejidad y que tienen implícitos en ellas la formulación de ideas, juicios y conceptos, así como la relación entre los mismos. (La autora)



## 2. Habilidades.

Las habilidades son acciones conscientes en los que interviene los procesos lógicos del pensamiento (análisis, síntesis, comparación, abstracción y generalización) y que favorecen el desarrollo de las capacidades. (11)

Es decir, son el dominio consciente y exitoso de la actividad. Su proceso de formación es complejo y está indisolublemente ligado a la formación de los conocimientos.

Existe un grupo de clasificaciones de habilidades, según el área en que se aplican:

### Clasificación de las habilidades

- ♣ **Habilidades profesionales:** a partir del **Modelo del Profesional** de cada carrera, se estructuran y definen las habilidades que debe tener el estudiante una vez egresado que le permitan su desempeño profesional, las cuales están estructuradas y se forman a nivel de colectivo de año, disciplina y asignatura. Estas habilidades responden a modos de actuación profesional para cada carrera. Entre ellas se destacan en la formación de docentes diagnosticar, planificar y ejecutar el proceso docente-educativo, evaluar, modelar, etc.;
- ♣ **Habilidades específicas:** son propias de las ciencias o de las tecnologías que son objeto de estudio o de trabajo en cada carrera y se concretan en los métodos de trabajo de cada disciplina y asignatura. Deben aparecer como contenido del programa. Pueden mencionarse por ejemplo: graficar, medir, diseñar, etc.
- ♣ **Habilidades lógicas o intelectuales:** contribuyen a la asimilación del contenido de las disciplinas y asignaturas y sustentan el pensamiento lógico tanto del aprendizaje como de la vida. Como las más comunes se pueden resaltar las siguientes: analizar, sintetizar, comparar, abstraer, explicar, aplicar, demostrar, argumentar, valorar, etc.;
- ♣ **Habilidades de comunicación:** propias del proceso docente, que son imprescindibles para su desarrollo, por ejemplo: tomar apuntes, hacer resúmenes, realizar lecturas rápidas y eficientes, en general, comunicar resultados por vía escrita u oral.



Las habilidades y los hábitos son resultado directo del proceso pedagógico y se forman durante el mismo, mientras que las capacidades se forman indirectamente. El proceso pedagógico influye directamente sobre las personas, induce su actuación mediante el planteamiento de objetivos y tareas, lo que provoca que el sujeto ponga en funcionamiento acciones y operaciones que bien dirigidas pueden llegar a sistematizarse y convertirse en habilidades y hábitos. Para ello es necesario que al plantear la tarea el profesor lo haga consciente de cuáles son las invariantes funcionales que el alumno debe realizar. Lo mismo sucede al plantear el objetivo; los alumnos deben conocerlo y hacerlo suyo para que éste contribuya realmente a la formación de habilidades. Las capacidades no se logran directamente en el proceso pedagógico pues a ellas las distingue su carácter de integración de conocimientos, habilidades, hábitos y otros procesos psicológicos de la personalidad.

Para que estas habilidades puedan ser desarrolladas es necesario tener en cuenta algunas **consideraciones sobre la estructura interna y el algoritmo** de las mismas

Para contribuir al desarrollo de habilidades en los estudiantes es preciso que el docente tenga en cuenta, de forma general, algunos aspectos: (12)

- ♣ Tanto el profesor como los alumnos deben conocer la estructura interna de las habilidades.
- ♣ Ejercitar el algoritmo, pues a través de la repetición el estudiante se apropia de su dinámica.
- ♣ Reflexionar sobre los diferentes pasos de la estructura interna y explicarlos, pues la palabra asociada a la acción da la solidez a la adquisición del algoritmo.
- ♣ Al iniciar el desarrollo de una habilidad el docente debe inducir al estudiante a dar los pasos por separado, controlar su asimilación y posteriormente dar todos los pasos de una sola vez, es decir, trabajar el sistema.
- ♣ Ejercitar sobre fuentes del conocimiento más sencillo que creen las bases para los más complejos.
- ♣ Ejecutar los pasos junto con el docente, luego realizarlos solos con la orientación del profesor y por último de forma independiente.





- ♣ Elevar gradualmente el nivel de asimilación de la habilidad de lo reproductivo a lo aplicativo.

Otro aspecto metodológico importante es la necesidad de propiciar situaciones de aprendizaje que permitan a los estudiantes, en dependencia de sus propios recursos, realizar las invariantes funcionales de una determinada ejecución del modo que le sea más cómodo y eficiente. Esto posibilita crear un ambiente de aceptación y confianza en el aula, permite la atención a las diferencias individuales.

Según lo antes expuesto, se considera que las habilidades son actividades concientes que realiza el individuo en las que intervienen un sistema de acciones, operaciones y conocimientos, que propician el desarrollo de capacidades. (La autora)

### **3. Habilidades Informáticas.**

Durante la concepción de la carrera y la determinación de las habilidades a formar se tienen en cuenta varios enfoques:

#### **Enfoque manual o instruccionalista. (Expósito C., Tesis doctoral (1988)) (8)**

Es el enfoque predominante en los inicios de la enseñanza de la programación, y se caracterizó por una enseñanza que utilizaba los elementos del recurso informático (en el código) y no en los procesos de búsqueda de solución de problemas.

#### **Enfoque algorítmico. (9)**

Se caracteriza por una enseñanza que hace el énfasis en métodos para elaborar algoritmos, en procedimientos algorítmicos y heurísticos para resolver problemas.

#### **Enfoque de proyecto.**

Se caracteriza por la subdivisión del proyecto en problemas parciales, necesarios y que motiven a la vez, el aprendizaje del nuevo contenido informático.

Debe su nombre a la tarea o problema general a resolver en el curso.



### **Enfoque de modelo.**

Se caracteriza por el uso de un programa o software que realiza una simulación

### **Enfoque de problema base.**

Se caracteriza porque se procede de forma inversa al enfoque del proyecto, cada modificación planteada convenientemente, al problema inicial (base), es un recurso que debe motivar la necesidad del nuevo conocimiento.

### **Enfoque problémico y heurístico.**

Se caracteriza por una enseñanza que hace énfasis en la necesidad del nuevo conocimiento informático a través de situaciones problémicas.

### **¿Cómo ha sido la enseñanza de la Informática hasta nuestros días?**

Las regularidades principales o formas regulares fundamentales de la enseñanza de la Informática son las siguientes: (4)

- ♣ La formación de conceptos,
- ♣ La elaboración de procedimientos y
- ♣ La resolución de problemas.

Las dos primeras, como las formas predominantes en la adquisición del conocimiento, y la tercera como la forma predominante, tanto para la creación de motivaciones adecuadas, como para la fijación de dicho conocimiento.

Por otra parte, la formación de conceptos informáticos se enmarca en el desarrollo del saber, mientras que la elaboración de procedimientos informáticos se enmarca, fundamentalmente, en el desarrollo del saber hacer; esencialmente en el desarrollo de habilidades tanto mentales como manipulativas o interactivas.

A continuación se abordan algunos elementos básicos, desde el punto de vista metodológico para que el profesor pueda planificar y ejecutar con éxito el proceso de enseñanza-aprendizaje de la Informática.



Toda concepción didáctica, por supuesto, debe tener en cuenta:

- ♣ Que el contenido es premisa o punto de partida de lo metodológico, y se enfatiza que cada uno de los conceptos informáticos básicos que se contemplen, como parte del curso, se identifica por su intención o características esenciales; esto es lo fundamental como contenido de enseñanza y punto de partida para la estructuración didáctica del concepto.
- ♣ La vía lógica elegida para la obtención del conocimiento como aspecto predominante en lo metodológico. En este sentido, tanto los conceptos como los procedimientos se pueden estructurar de lo particular a lo general; es decir, del análisis de casos particulares inferir lo general, (vía inductiva) o a la inversa de lo general a lo particular (vía deductiva).
- ♣ Planificar cuidadosamente el conjunto de actividades necesarias para la fijación del conocimiento.

Esta es una cuestión de gran significación si tenemos en cuenta que la preparación del alumnado para la resolución de problemas, haciendo uso de los medios informáticos y de cómputos disponibles en la escuela, debe ser uno de los objetivos priorizados de la inserción curricular de la Informática.

Por último, el profesor debe dominar los enfoques didácticos predominantes en la enseñanza de la Computación, que son procedimientos didácticos que pueden ser aplicados en la enseñanza de la Informática, válidos tanto, para la orientación pedagógica de todo un curso, como para la enseñanza de un contenido específico en una clase o parte de ella. Entre los enfoques más representativos tenemos:

- El del Manual o Instruccionista,
- El Algorítmico,
- El Problémico,
- El enfoque del Modelo,
- El Enfoque del Proyecto,



- Y del Problema Base.

Como componentes de los métodos de enseñanza, ninguno de estos enfoques, por sí solo, debe ser el que se aplique en términos absolutos en la enseñanza de la Computación. La preparación que haya alcanzado cada profesor, en el orden didáctico, le guiará en la toma de decisión de cuál sería el más conveniente o necesario a aplicar en cada momento o etapa del curso.

Al precisar los criterios de varios autores encontramos que:

(9) Relativo a la enseñanza de la Informática señala tres aspectos fundamentales:” la formación de conceptos, la elaboración de procedimientos, y la resolución de problemas”. Algunos elementos de Metodología de la Enseñanza de la informática (libro). Expósito C., (2001)

Concibe el ambiente de enseñanza aprendizaje de la informática basado en diferentes teorías pedagógicas (asociacionistas, constructivistas, ambivalentes, contextual. Bartolomé, A. 1998.España

“No se dispone aún de un sistema didáctico que constituya una vía para la enseñanza de la computación. Ha existido un desarrollo acelerado de la computación sin un tratamiento didáctico”. González E., (1995)

“...el desafío histórico no consiste en difundir lenguajes de programación, sino concentrarse en las habilidades.” González E., (1995)

“No se dispone aún de un sistema didáctico que constituya una vía para la enseñanza de la computación. Ha existido un desarrollo acelerado de la computación sin un tratamiento didáctico”. Rodríguez, R., (1996),

“la falta de integración teórica – metodológica en el campo de la Didáctica de la Informática”. González Hernández, W. (2001)



“...la introducción de las líneas directrices y los núcleos temáticos conceptuales abren nuevas perspectivas teóricas y metodológicas para la enseñanza de la informática”. González Hernández, W. y Estrada S. V. (2002)

“... aún no existe una metodología propia de la enseñanza de esta disciplina que tenga presente todos los requerimientos educativos e instructivos”. Fernández Gutiérrez, F. ,(2001)

Al analizar todos los aspectos abordados en este epígrafe podemos decir que

Una habilidad informática son actividades conscientes que realiza el individuo en las que intervienen un sistema de acciones, operaciones y conocimientos, que propician el desarrollo de capacidades y que van encaminadas al uso y explotación adecuada del Hardware y el Software. (La autora)

#### **4. Un hiperentorno de aprendizaje para la escuela cubana. El software educativo.**

Actualmente, existe y se consolida un modelo de enseñanza en el que la informática ocupa un lugar bien definido. Este modelo está estrechamente relacionado con el entorno tecnológico donde la sociedad se desarrolla, además el mismo se encuentra en constante evolución.

El análisis de todo ello, de forma integral, nos permite considerar que la computadora y los materiales de estudio computarizado, entiéndase software educativos, utilizados por el profesor, coinciden con cada uno de estos elementos incluidos en la definición. Es decir es un dispositivo de cuyo uso se puede derivar una reconceptualización de la enseñanza, propicia un conocimiento por diferentes vías relacionadas precisamente con la naturaleza de la misma.

La computadora y los softwares educativos, como medios de enseñanza resultan un eficiente auxiliar del profesor en la preparación e impartición de las clases ya que contribuyen a una mayor ganancia metodológica y a una racionalización de las actividades del profesor y los alumnos.



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, etc.**

#### **4.1. Propuesta de una arquitectura de confección de software educativo. (16)**

La arquitectura de un software se caracteriza por su flexibilidad a la hora de seleccionar sus componentes y que en general está compuesta por los siguientes estilos:

##### **Hipermedia**

Constituye un soporte bibliográfico de nuevo tipo. Garantiza una lectura no lineal que se ramifica a partir de anclas multimedia que no son más que elementos interactivos que generan hiperenlaces con nuevos componentes informativos, que paulatinamente conforman diferentes niveles de asimilación, a la vez que de manera subliminal generan mapas conceptuales personalizados auspiciando así la atención de las características individuales de los educandos.

Hoy en día este tipo de lectura no lineal constituye la base fundamental de las estrategias contemporáneas de presentación de la información en los modernos soportes de almacenamiento de la información digital y constituye un invariante que debe desarrollarse desde las edades más tempranas, conjuntamente con el desarrollo de habilidades de búsqueda y selección.

##### **Servicios informáticos posibles:**

- ♣ Navegación.
- ♣ Localización.
- ♣ Hiperenlaces.



- ♣ Selección y exportación.
- ♣ Impresión.
- ♣ Búsqueda plena a través de palabra o frase clave.
- ♣ Acceso a otros módulos.

### **Entrenador-Repasador**

Está caracterizado por un alto nivel de interactividad. Puede materializarse a través de sistemas de cuestionarios, planteamientos de problemas a resolver, evaluaciones de control del aprendizaje, empleo de simuladores, etc. Tiene un carácter eminentemente práctico.

#### **Servicios informáticos posibles:**

- ♣ Retroalimentación MM.
- ♣ Evaluación del desempeño.
- ♣ Control paralelo del trabajo en pareja.
- ♣ Presentación lineal o aleatoria.
- ♣ Establecimiento de una estrategia de enseñanza–aprendizaje.
- ♣ Control de efectividad, tiempo de ejecución y de intentos de acuerdo con la estrategia definida.
- ♣ Acceso a otros módulos.

### **Glosario especializado**

Diccionario terminológico, opcionalmente ilustrado.

#### **Servicios informáticos posibles:**

- ♣ Definición.
- ♣ Apoyo MM.
- ♣ Análisis ortográfico.
- ♣ Mecanismos de búsqueda.
- ♣ Impresión.
- ♣ Acceso a otros módulos.



## **Galería**

Módulo que concentra los recursos multimedia (sonido, imagen fija, videos, gráficos, etc.), distribuidos a través del todo el programa.

### **Servicios informáticos posibles:**

- ♣ Acceso puntual a componentes MM.
- ♣ Puede contener referencias a esquemas, mapas, gráficos, formularios, etc.
- ♣ Mecanismos de búsqueda.
- ♣ Acceso a otros módulos.

## **Tutor**

Simulador de la actividad del proceso de introducción de contenidos. Secuencias de voz o imagen del maestro o locutor sincronizada con elementos MM (esquemas, textos, imágenes, diagramas, fórmulas, etc.)

### **Servicios informáticos posibles:**

- ♣ Acceso por secciones, capítulos, etc.
- ♣ Repetición.
- ♣ Navegación.
- ♣ Pueden emplearse elementos o preguntas que permitan materializar una estrategia de presentación de la información similar a como hace un maestro a través de diálogos socráticos en el aula.
- ♣ Marcadores.
- ♣ Mecanismos de búsqueda.
- ♣ Impresión.
- ♣ Acceso a otros módulos.

## **Componente lúdico**

Es el juego instructivo en el que se presenta como esencia la idea de aprender jugando. Se caracteriza por la existencia de situaciones metaforizadas en las que existen retos, premios, la combinación de la habilidad con lo fortuito, existencia de niveles de complejidad, etc.





### **Servicios informáticos posibles:**

- ♣ Control de resultados de los jugadores.
- ♣ Almacenamiento de estados del juego.
- ♣ Carácter multiusuario.

### **Traza**

Constituye el registro de la actividad del estudiante.

### **Servicios informáticos posibles:**

- ♣ Identificación del estudiante.
- ♣ Fecha y hora de comienzo de la actividad.
- ♣ Duración de la actividad.
- ♣ Contenidos abordados.
- ♣ Ejercicios realizados, respuestas del alumno y errores cometidos.
- ♣ Índice de efectividad de la tarea.
- ♣ Tiempo consumido en cada actividad.
- ♣ Valoración general de la actividad.

### **Esquina del maestro**

Posibilidad de crear una biblioteca virtual para el maestro con documentación científico-metodológica actualizada.

Ejemplos pueden ser:

- ♣ Recomendaciones metodológicas para el tratamiento del tema o para el uso del programa como medio de enseñanza.
- ♣ Referencias a otros programas educativos que puedan complementarse con el presentado.
- ♣ Artículos de carácter científico-metodológico sobre el tema.



### **Servicios informáticos posibles:**

- ♣ Navegación.
- ♣ Localización.
- ♣ Hiperenlaces.
- ♣ Selección y exportación.
  
- ♣ Impresión.
- ♣ Búsqueda plena a través de palabra o frase clave.
- ♣ Glosario.

Para lograr que el aprovechamiento de las computadoras en el proceso docente tenga un papel relevante, se hace necesario dotarlas de un software educativo de calidad, lo que debe medirse en términos del conocimiento que sean capaces de representar y transmitir. La arquitectura que hemos presentado puede contribuir al logro de este propósito si hacemos una correcta selección y utilización de los módulos que se necesiten para contribuir a resolver los problemas detectados en el proceso de enseñanza–aprendizaje y los enriquezcamos a partir de la experiencia obtenida de su utilización.

De lo anterior podemos deducir que el software educativo es aquel programa destinado a contribuir con el eficiente desarrollo del proceso de enseñanza y aprendizaje. (La autora)

### **5. Un entrenador.**

Un entrenador es el software diseñado con el propósito de contribuir al desarrollo de una determinada habilidad, intelectual, manual o motora, en el estudiante que lo utiliza. (17)

#### **Sistemas entrenadores.**

Designamos con este nombre al software educativo diseñado con el propósito de contribuir al desarrollo de una determinada habilidad, intelectual, manual o motora, en el estudiante



que lo utiliza por lo que profundizan en las dos fases finales del aprendizaje: aplicación y retroalimentación.

Se parte de que los estudiantes cuentan con los conceptos y destrezas que van a practicar.

En este tipo de material deben conjugarse diferentes aspectos:

- 1.- Cantidad de ejercicios
- 2.- Variedad en los formatos
- 3.- Niveles en los ejercicios
- 4.- Selección de ejercicios
- 5.- Motivación
- 6.- Creación de expectativas
- 7.- Sistema de refuerzo y retroinformación.
- 8.- Retroalimentación

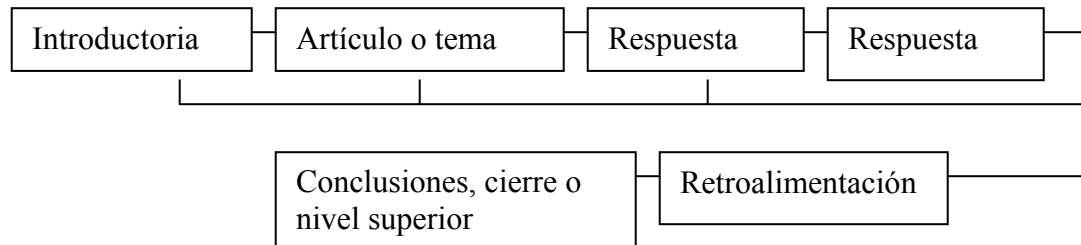
Como se conoce, la fijación de la información se basa en la repetición de la práctica, esto último es lo que permite al estudiante alcanzar las habilidades necesarias proporcionándole entre otros: facilidad, seguridad y velocidad en la interpretación y resolución de los problemas planteados.

Por ello estos programas poseen gran potencial para incrementar la eficiencia y la efectividad de un entrenamiento ya que permiten enfatizar la práctica en ejercicios en los cuales el estudiante puede tener determinada dificultad para resolver, cosa que no es posible en los manuales de práctica. Además, permiten clasificar los ejercicios por dificultad y brindan la posibilidad de que el estudiante comience por los ejercicios más fáciles y mientras se entrena va aumentando el grado de dificultad de los ejercicios.

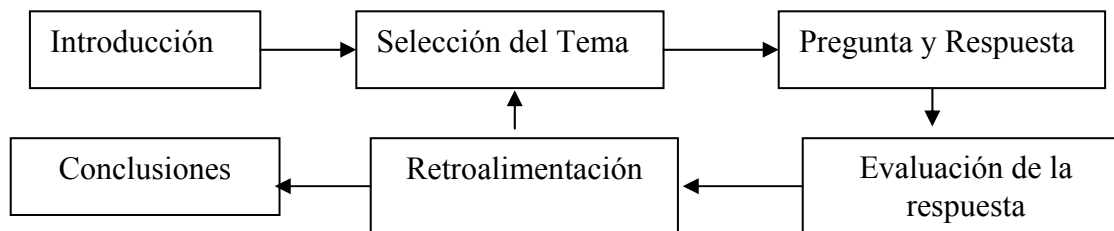
También permiten el desarrollo de determinados tipos de habilidades, donde el estudiante tiene el control de todas las acciones; en él no se realiza una conducción del proceso aprendizaje, pues el alumno decide la tarea en la que desea entrenarse.



A continuación se muestra una probable estructura de un entrenador (17)



Aunque otros autores (12) proponen la siguiente alternativa como esquema:



En la sección introductoria se debe considerar:

- ♣ La forma de controlar al estudiante en su actuación.
- ♣ Analizar si este estudiante ha tenido una actuación anterior y cómo fue la misma.

Hay tres procesos que se dan por ligados:

1) La selección del artículo o tema, cuyas características son:

- ♣ El tipo de tema que se va a desarrollar.
- ♣ La utilización de gráficos, imágenes, animaciones, etc.
- ♣ Nivel de dificultad del tema en cuestión teniendo presente conservar el nivel de dificultad constante, basarlo en el desarrollo del estudiante y en la misma medida que vaya venciendo un nivel ir incrementando los mismos.
- ♣ Mantener una secuencia lógica de pasos para pasar de un tema a otro.
- ♣ Considerar una forma de generación a través de un algoritmo que permita obtener los temas.



Para seleccionar los temas se pueden considerar si los mismos van a obtenerse aleatoriamente o tomados de una cola organizada o algún otro método que nos permita esta selección.

2) Para establecer las preguntas y juzgar las respuestas dadas por el estudiante se deben tener en cuenta los mismos principios que para un tutorial siempre pensando que estamos desarrollando un sistema entrenador y que por tanto varían los objetivos.

En el caso de la retroalimentación hay que profundizar en el aspecto motivacional del estudiante el cual puede estar dado por:

- ♣ La competencia que a su vez puede ser contra otros estudiantes, contra la computadora, contra él o contra reloj.
- ♣ Empleo de múltiples modos de pantallas y variedad de las mismas.
- ♣ Refuerzos adjuntos (por ejemplo otras actuaciones dentro del entrenamiento).
- ♣ Longitud o tiempo de la sesión de entrenamiento, la cual no debe exceder los 45 minutos, considerándose por muchos especialistas que para los niños estas deben durar como máximo 25 minutos y para el resto entre 30 y 45 minutos.

En la última etapa, puede ocurrir un cierre del entrenador, en este caso, la actuación del estudiante debe ser almacenada como constancia de su desarrollo por el mismo, además de darse las conclusiones. Igualmente puede ocurrir que se pase a otro nivel de dificultad de acuerdo a lo que está programado.

Los entrenadores presentan un enfoque algorítmico. Puede decirse que bajo este enfoque se da una educación “controlada por el diseñador”. Estos deben contar con un módulo “maestro” o “entrenador” que también debe ser capaz de identificar y caracterizar al estudiante que lo emplea y seguir su estrategia de entrenamiento de acuerdo a sus capacidades y programa.

Otra de sus características es la base de conocimientos del entrenador que debe incluir un conjunto estructurado de ejercicios o tareas que deben ser presentadas por el entrenador las cuales deben estar convenientemente relacionadas con los conocimientos de la base, de



forma que el entrenador sea capaz también de auxiliar al estudiante con el conocimiento necesario para solucionar cada ejercicio.

Resumiendo se llega a la idea de que un entrenador es un software educativo que facilita el desarrollo de habilidades y capacidades del usuario que lo utilice. (La autora)



## **Epígrafe 2. Ingeniería del software aplicado al recurso.**

Durante el desarrollo del curso escolar (2004-2005), los profesores que imparten la asignatura Proyectos Informáticos, se enfrentaron a serias dificultades en cuanto a la asimilación de los contenidos por parte de los alumnos, principalmente los de primer año. Después de un diagnóstico de las posibles causas que propiciaban esta situación, se llega a la conclusión de que existe un insuficiente desarrollo del pensamiento lógico en los estudiantes del politécnico de informática: "Carlos Hidalgo Díaz" del municipio de Pinar del Río.

Al analizar el resultado de los instrumentos aplicados, derivamos como principal

### **Problema:**

¿Cómo contribuir al desarrollo del pensamiento lógico de los estudiantes del Politécnico de Informática: "Carlos Hidalgo Díaz", en el proceso de enseñanza de la asignatura Proyectos Informáticos?

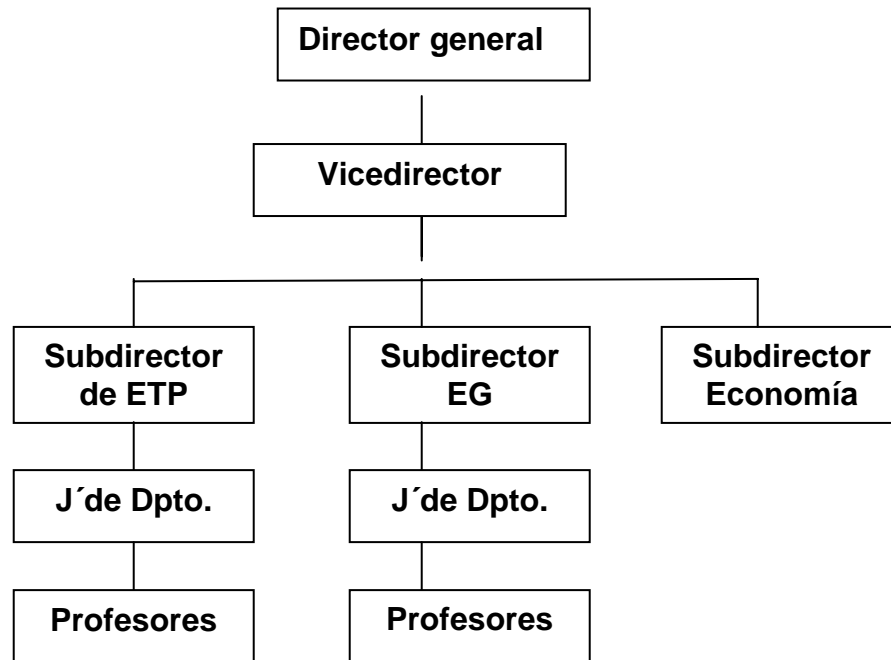
En el Politécnico de Informática: "Carlos Hidalgo Díaz", se lleva a cabo un proceso de enseñanza y aprendizaje a gran escala, por la cantidad de alumnos que en él estudian y por las características de que tienen la gran mayoría de los profesores (profesores en formación), de ahí que se le dedique gran atención a la preparación de los mismos, y a la de los alumnos, principalmente a los de primer año.

### **Su misión es lograr:**

- ♣ Insertar a los alumnos en el desarrollo de proyectos. Como programadores.
- ♣ Formar hombres comprometidos con la patria.



Su estructura organizacional se muestra en la siguiente figura:



Esta escuela se ha tomado como modelo para el levantamiento de información y generalización de la problemática sobre el desarrollo del pensamiento lógico de los alumnos, que puede ser común a otras escuelas.

El proyecto propone, por tanto, desarrollar una aplicación que permita el desarrollo del pensamiento lógico de los alumnos a través de la solución de ejercicios y mediante el control de sus resultados, permitir a los profesores la valoración de este aprendizaje, así como la manipulación de los datos relacionados con los alumnos y su aprendizaje y más aún el poder gestionar el banco de ejercicios implementado en el recurso, aspirando a que pueda ser aplicada a otras escuelas con características similares.

El proyecto abarca los siguientes procesos:

- ♣ Autenticación: El usuario se identifica para poder acceder al software
- ♣ Resolución de ejercicios: Partiendo de la selección del tema en que quieren entrenarse, el estudiante accede a los ejercicios y tiene la posibilidad de retroalimentar los contenidos necesarios para la solución de las problemáticas planteadas.





- ♣ Valoración del aprendizaje del los estudiantes: El profesor puede visualizar los datos relacionados con los alumnos y su aprendizaje y a partir de ahí, manipular esta información para ser visualizada a su conveniencia.
- ♣ Gestión de ejercicios: El profesor podrá eliminar, modificar o incrementar la base de ejercicios propuestos.

En este software se encontrará una serie de ejercicios encaminados a la búsqueda de vías de solución según la lógica, dirigidos a desarrollar habilidades específicas de la asignatura, relacionados con problemáticas de la vida práctica y variados, los cuales se reforzarán y tendrán un apoyo de contenidos sintetizados, imágenes, animaciones, videos, así como de una ayuda que le facilitará la navegación por la aplicación, un glosario de palabras que pueden ser difíciles de entender y la bibliografía a la que pueden dirigirse para ampliar los conocimientos relacionados con los temas que se tratan.

De ahí que nuestro principal **objetivo** sea:

Entrenar a los alumnos de primer año del Politécnico de Informática: “Carlos Hidalgo Díaz” a través de un sistema de ejercicios que contribuyan al desarrollo del pensamiento lógico de los mismos.

Como **objetivos específicos** se plantean los siguientes:

1. Desarrollar el pensamiento lógico de los alumnos.
2. Crear un banco de ejercicios y contenidos, tratados de forma amena.
3. Digitalizar el control del aprendizaje de los alumnos en este tipo de problemática.

### **Funcionalidades previstas.**

Para cumplir con los objetivos propuestos se prevé que el sistema tenga las siguientes funcionalidades:



- ♣ Visualizar información.
- ♣ Presentar ejercicios.
- ♣ Promediar resultados.
- ♣ Almacenar datos.
- ♣ Mostrar datos según la selección.
- ♣ Enviar mensajes.
- ♣ Gestionar usuarios del sistema de acuerdo a los niveles de acceso. (Ver su relación con categorías, atributos y detalles en Anexo XIX)

### **Reglas de negocio.**

Cuando un profesor imparte una clase, propone al alumno un conjunto de ejercicios a resolver y a través de la revisión de la solución del mismo, se controla su aprendizaje, el cual se registra en los documentos del profesor y al final se hacen los cálculos necesarios.

Con la utilización del sistema que se propone, el profesor puede tratar estos ejercicios sin necesidad de hacer cálculos para el control del aprendizaje, pues el propio sistema los realiza y en función de la puntuación dada, define el nivel del aprendizaje en que se encuentra. Además el mismo puede ser utilizado fuera de la clase.

El producto incluirá módulos. Uno principal que contendrá una breve explicación, dando paso a un diagnóstico inicial del alumno y que después de realizado el alumno podrá seleccionar en otro, la asignatura que quiere tratar, dando paso a otro menú por temas de las asignaturas, en ellas encontrará un sistema de ejercicios que de no saber sus respuesta puede acceder a los módulos de contenidos, imágenes, animaciones, videos, bibliografía y por último la ayuda.



**Público al que va dirigido:**

Enseñanza: ETP.

Institución: Politécnicos de Informática.

Año: primero.

Edades: 15 y 16 años.

**Cliente:**

Profesores

**Usuarios:**

Profesores

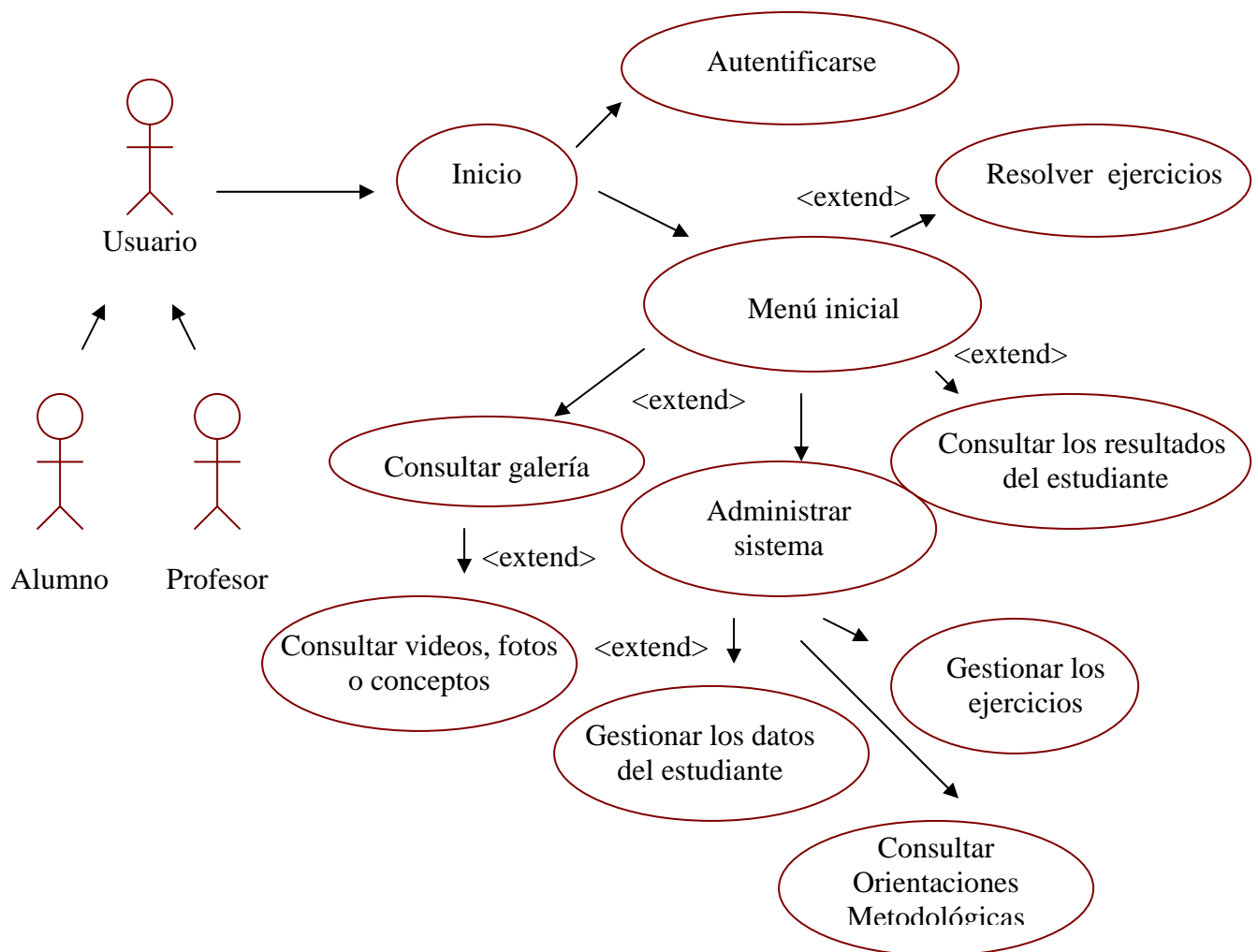
Alumnos



### Prerrequisitos:

Uso del mouse y el teclado

### Diagrama de casos de uso del negocio.



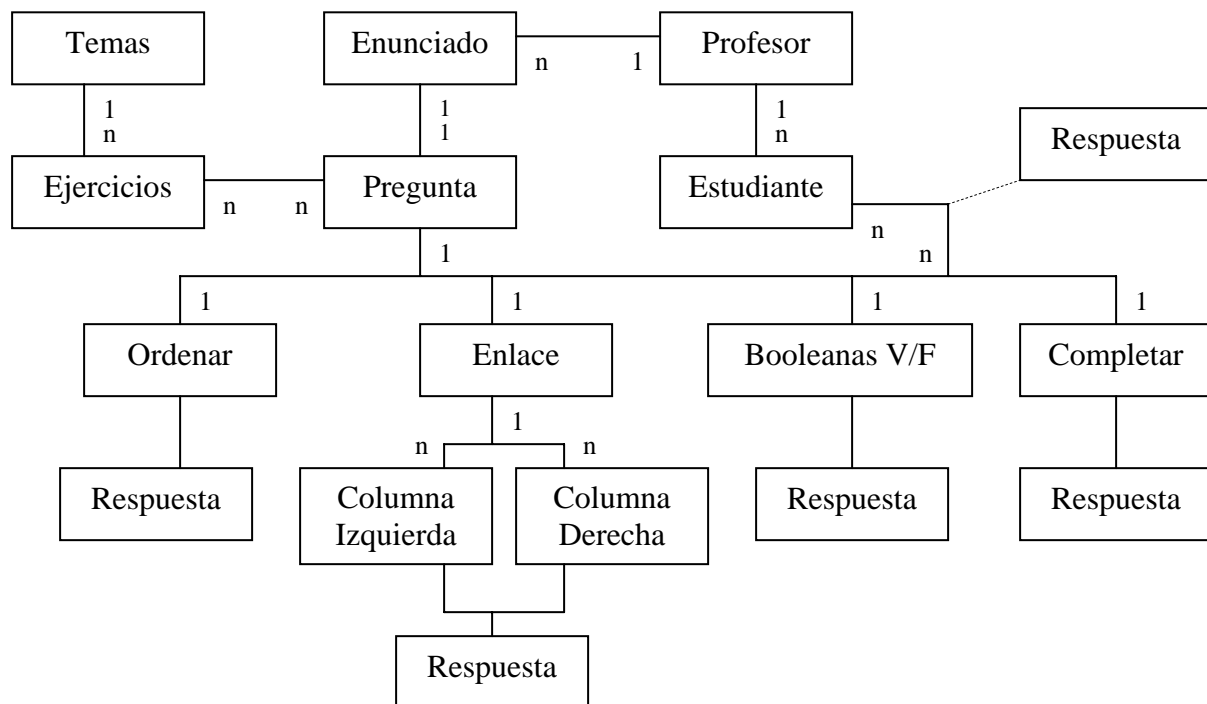
### Diagrama conceptual del proyecto.

En este sistema pueden participar los alumnos de primer año de la escuela, los cuales podrán seleccionar el tema de la asignatura que desean evaluar, que no es más que aquellos que aparecen en los programas de estudio. Cada uno de estos ejercicios tendrá un enunciado, el cual hará referencia a una situación problemática en específico y una pregunta en la cual se orientará las acciones a llevar a cabo para dar respuesta al ejercicio. Las preguntas estarán redactadas en



correspondencia con el tipo de ejercicio que se quiere desarrollar: Ordenar: Se darán las posibles respuestas de forma desordenada y el alumno debe establecer el orden correcto; Enlace: Existe una columna izquierda en la que aparecen los conceptos y en la columna derecha aparecen las definiciones o sinónimos con los que se identifican los elementos de la columna anterior y con los que deben ser relacionados; verdaderos o Falsos: Se dan posibles respuestas y se selecciona la que está acorde con el enunciado y la pregunta y Completar: Se expresan ideas o conceptos de forma incompleta, dejando el espacio en blanco según el lugar que corresponda y en él el alumno debe escribir la palabra correcta. A cada una de estas preguntas da su propia respuesta.

En el caso del profesor, que no es más que aquel que actuará como administrador del sistema, podrá realizar los mismos ejercicios hechos por el alumno o consultará los datos relacionados con los mismos.



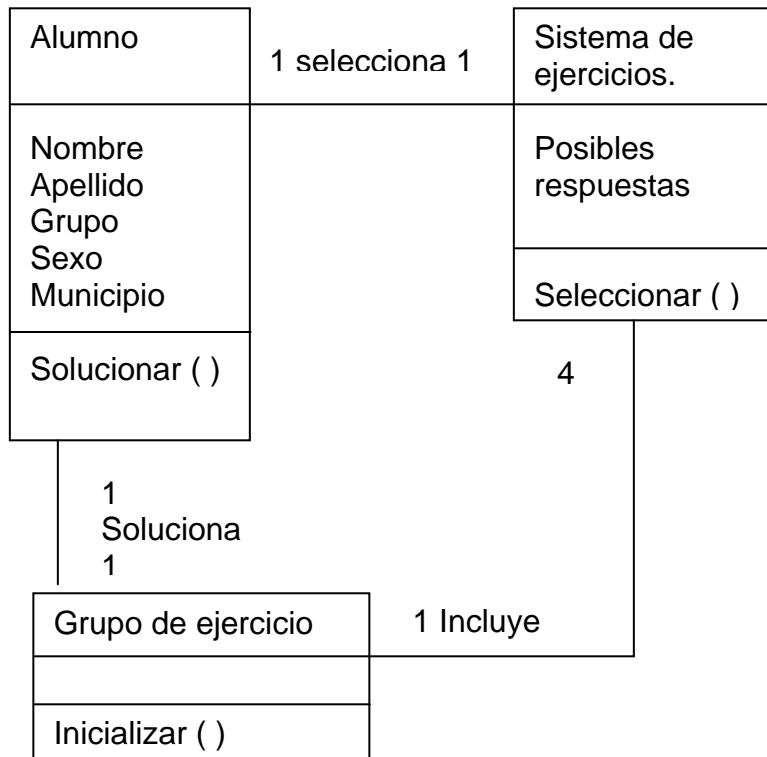


### Realización de los casos de uso del negocio.

CASO DE USO DEL SOFTWARE		Solución de ejercicios por parte del alumno.
Actores	Alumno (Inicia)	
Propósito	Registrar los resultados del aprendizaje de cada alumno, partiendo de la solución de los ejercicios y la consulta de materiales específicos de cada asignatura.	
<b>Resumen:</b> El caso de uso se inicia cuando el alumno accede al software, se identifica y selecciona el tema de la asignatura en la cual quiere entrenarse, para dar paso a la solución de los ejercicios relacionados con los mismos. En dependencia de la respuesta escogida recibirá una puntuación que al final se promedia y se almacena. El usuario tendrá la opción de acceder a los contenidos que se relacionan con el tema escogido.		
Curso normal de los eventos.		
Acción del actor		Respuestas del sistema.
1. El caso de uso comienza cuando el usuario ejecuta el software.		2. Se presenta una ventana de diálogo con datos personales a llenar.
3. El usuario se identifica.		4-a. Se almacenan los datos. 4-b. Se presenta la ventana principal con la relación de asignaturas a trabajar.
5. El usuario escoge la opción de resolver ejercicios.		6. Muestra el test.
7. Selecciona el tema a tratar.		8. Se visualizan los ejercicios con las diferentes posibles respuestas.
9. Selecciona la respuesta correcta en cada ejercicio.		10-a. Asigna una puntuación por cada respuesta dada. 10-b. Almacena los datos. 10-c. Genera un mensaje de error o felicitación en dependencia de la respuesta dada. Si el mensaje es de error da la opción de volver a responder y de acceder a la ayuda.



### Diagrama de clases.





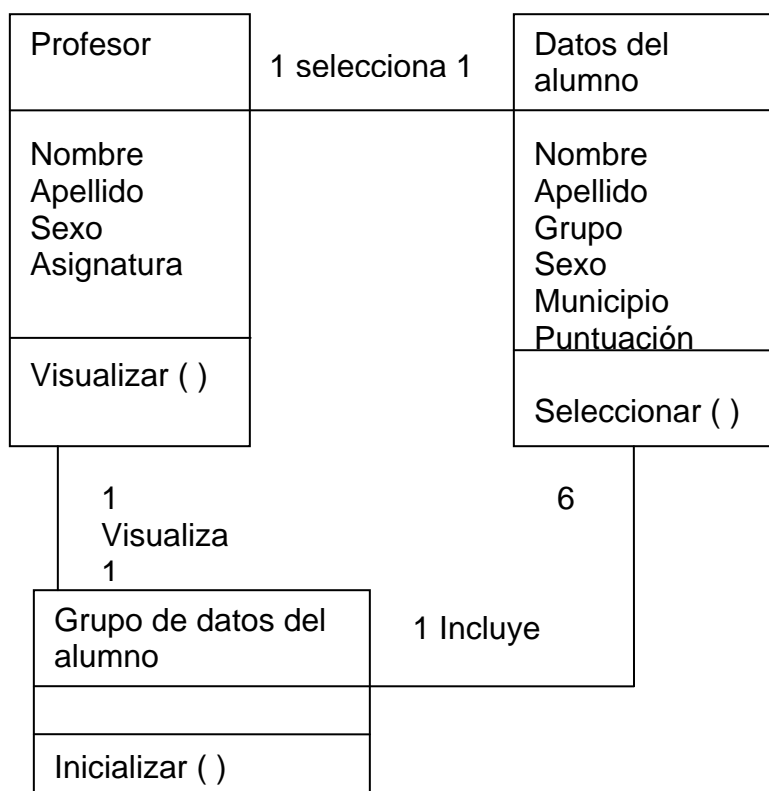
CASO DE USO DEL SOFTWARE		Valoración del aprendizaje del alumno por parte del profesor.
Actores	Profesor (Inicia)	
Propósito	Valorar los resultados del aprendizaje de cada alumno, partiendo de la visualización y manipulación de la base de datos.	
Resumen:		
El caso de uso se inicia cuando el profesor accede al software, se identifica y selecciona el acceso a la base de datos como opción. En este caso aplicará los filtros deseados para visualizar los datos necesarios. Si quiere imprimirá los datos extraídos del informe.		
Curso normal de los eventos.		
Acción del actor		Respuestas del sistema.
1. El caso de uso comienza cuando el usuario ejecuta el software.		2. Se presenta una ventana de diálogo con datos personales a llenar.
3. El usuario se identifica.		4-a. Se almacenan los datos. 4-b. Se presenta la ventana principal con las opciones a realizar
5. El usuario escoge la opción de visualizar los datos		6. Muestra la base de datos
7. Aplica los filtros		8. Muestra la información
9. Escoge la opción de elaborar el informe y luego imprimir de forma opcional		10. Muestra el informe elaborado e imprime
11. Selecciona el tema a tratar.		12. Se visualizan los ejercicios con las diferentes posibles respuestas.
13. Salir del software		14. Cierra el programa
Prioridad		Responde al principal objetivo de automatización al





	resolver parte del problema.
Mejoras	Datos que contiene el producto
Otras secciones	

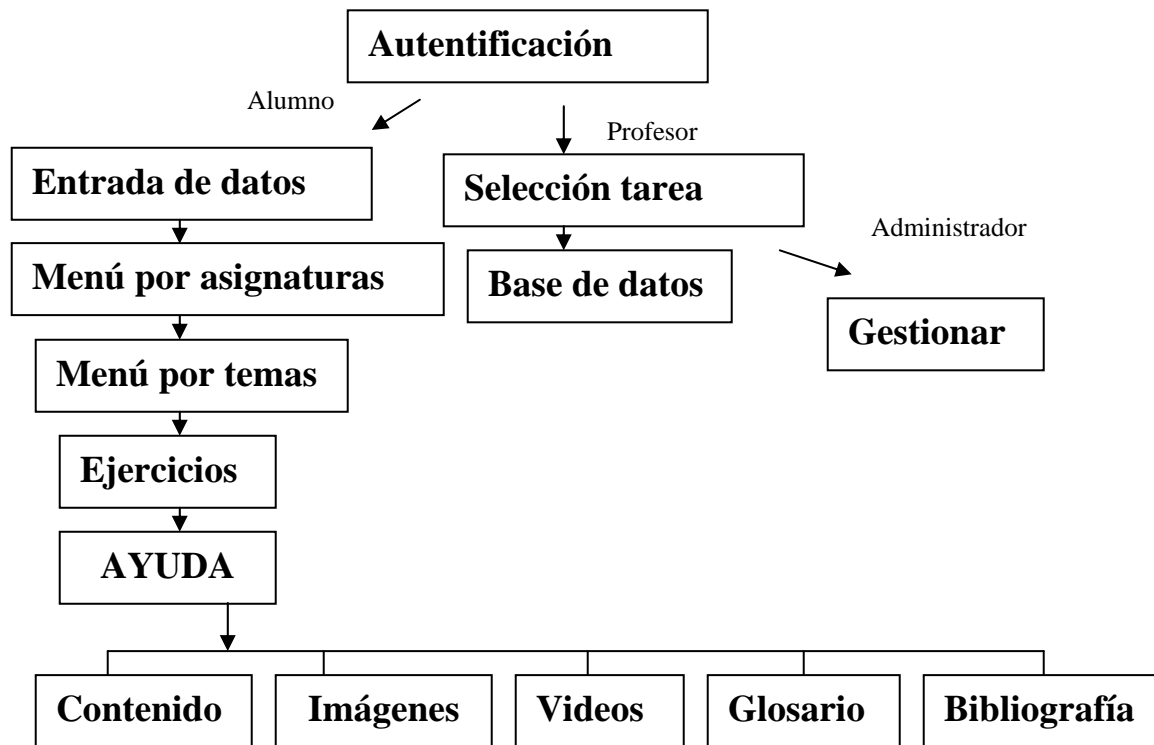
### Diagrama de clases.



(Ver el resto de los casos de usos en el Anexo XX)



**Posible estructura modular:**



La descripción de cada pantalla, con la visualización de su interfaz aparece en el Anexo XVI.



### Análisis de Factibilidad

Existe un grupo de dificultades, en el IPI: "Carlos Hidalgo Díaz", en función de la preparación de los profesores que impide que los alumnos puedan desarrollar su pensamiento lógico, al realizar su valoración vimos que se necesita la instrumentación de un sistema que pueda ser utilizado como medio de enseñanza y que se adecue a las exigencias planteadas por nuestros clientes y usuarios. Se comenzó calculando aproximadamente el costo del mismo con el uso del Modelo de Diseño temprano de COCOMO II usando como métrica la de los puntos de función. En la figura I.3.1 puede ver el cálculo de líneas de código fuente según los Puntos de Función, considerando como lenguaje de desarrollo Unix Shell Script, ya que es el que más se asocia al que en realidad se utiliza: el open script de la aplicación ToolBook, obteniendo **2583** líneas fuentes.

**SLOC Input Dialog - El pensamiento**

Sizing Method:  
☐ SLOC  
☒ Function Points  
☐ Adaptation and Reuse

Breakage  
% of code thrown away due to requirements evolution and volatility  
REVL

Module Size in Function Points  
Language  Change Multiplier

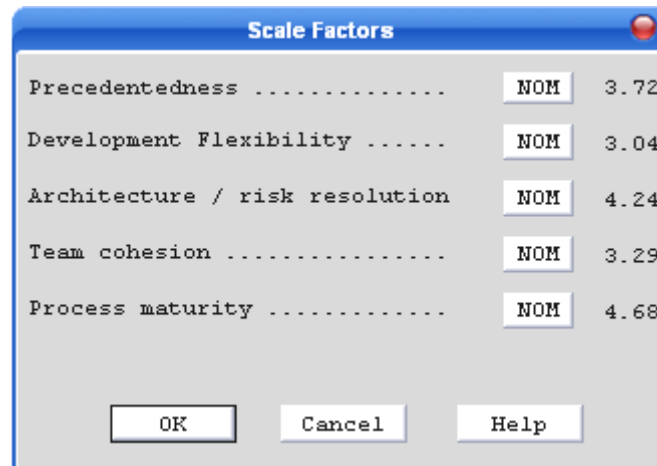
Function Type	# of Function Points			SubTotal
	Low	Average	High	
Internal Logical Files	<input type="text" value="0"/>	<input type="text" value="7"/>	<input type="text" value="0"/>	70
External Interface Files	<input type="text" value="4"/>	<input type="text" value="0"/>	<input type="text" value="0"/>	20
External Inputs	<input type="text" value="3"/>	<input type="text" value="0"/>	<input type="text" value="0"/>	9
External Outputs	<input type="text" value="2"/>	<input type="text" value="0"/>	<input type="text" value="0"/>	8
External Inquiries	<input type="text" value="0"/>	<input type="text" value="4"/>	<input type="text" value="0"/>	16
Total Unadjusted Function Points				123
Equivalent Total in SLOC				2583

OK Cancel Help

**Figura I.3.1 Cálculo de líneas código fuente.**



Los valores tomados para los Factores de Escala (SF<sub>j</sub>) fueron:



Scale Factors		
Precedentedness .....	NOM	3.72
Development Flexibility .....	NOM	3.04
Architecture / risk resolution	NOM	4.24
Team cohesion .....	NOM	3.29
Process maturity .....	NOM	4.68

OK Cancel Help

**Figura I.3.2 Factores de Escala.**

Siendo:

- ♣ **Precedentedness:** Desarrollo de Software previos similares al actual.
- ♣ **Development Flexibility:** Flexibilidad en el desarrollo.
- ♣ **Architecture / risk resolution:** Arquitectura y Manejo de riesgos.
- ♣ **Team cohesion:** Cohesión del equipo de desarrollo.
- ♣ **Process naturity:** Nivel de Madurez del software



Los valores tomados de los Factores de Multiplicidad del Esfuerzo para el Modelo de Diseño Temprano son:

base + Incr % = rating

Product:	RELY	DATA	DOCU	CPLX	RUSE
base	NOM	NOM	NOM	NOM	NOM
Incr%	0%	0%	0%	0%	0%

Platform:	TIME	STOR	PVOL
base	NOM	NOM	NOM
Incr%	0%	0%	0%

Personnel:	ACAP	PCAP	PCON	APEX	LTEX	PLEX
base	NOM	NOM	NOM	NOM	NOM	NOM
Incr%	0%	0%	0%	0%	0%	0%

Project:	TOOL	SITE
base	NOM	NOM
Incr%	0%	0%

User:	USR1	USR2
base	NOM	NOM
Incr%	0%	0%

EAF is also affected by Schedule

EAF: 1.00

OK Cancel Help

**Figura I.3.3 Valores de los Multiplicadores de Esfuerzo**

Donde:

- ♣ **RCPX:** Confiabilidad y complejidad del producto.
- ♣ **RUSE:** Nivel de reutilizabilidad del desarrollo.
- ♣ **PDIF:** Dificultad de uso de la plataforma.
- ♣ **PERS:** Capacidad del personal de desarrollo.
- ♣ **PREX:** Experiencia del personal de desarrollo.
- ♣ **FCIL:** Facilidades de desarrollo.
- ♣ **SCED:** exigencias sobre el calendario.



El resultado obtenido fue:

Total Lines of Code:	2583	Estimated	Effort	Sched	PROD	COST	INST	Staff	RISK
		Optimistic	6.7	6.7	386.8	1666.84	0.6	1.0	
		Most Likely	8.3	7.2	309.4	2083.56	0.8	1.2	0.0
		Pessimistic	10.4	7.7	247.5	2604.44	1.0	1.3	

**Figura I.3.4 Resultado de la estimación**

Siendo:

- ♣ **Effort:** Esfuerzo (Hombres-Mes)
- ♣ **Sched:** Tiempo (Meses)
- ♣ **Prod:** Productividad (Instrucciones/Hombre-Mes)
- ♣ **Cost:** Costo (unidad monetaria)
- ♣ **Staff:** Personal (hombres)
- ♣ **Risk:** Riesgo (solo valido en el Modelo Post Arquitectura)

Dando de cada indicador tres valores:

- ♣ **Optimistic:** valor optimista
- ♣ **Most Likely:** valor esperado
- ♣ **Pessimistic:** valor pesimista

El valor de cada indicador se obtuvo mediante una media ponderada de los valores dados:

$$[\text{Valor Optimista} + 4 * (\text{Valor Esperado}) + \text{Valor Pesimista}] / 6$$



Obteniendo los valores siguientes:

**Esfuerzo:**

$$[6.7 + 4 (8.3) + 10.4] / 6 = \mathbf{8.38 \text{ H-M}}$$

**Tiempo de Desarrollo:**

$$[6.7 + 4 (7.2) + 7.7] / 6 = \mathbf{7.2 \text{ Meses}}$$

**Productividad:**

$$[386.8 + 4 (309.4) + 247.5] / 6 = \mathbf{311.98 \text{ PM}}$$

**Costo:**

$$[1666.84 + 4 (2083.56) + 2604.44] / 6 = \mathbf{\$2100.92 \text{ Unidad Monetaria}}$$

Resumiendo el valor de cada indicador tenemos:

<b>Esfuerzo</b>	<b>Tiempo de desarrollo</b>	<b>Productividad</b>	<b>Costo</b>
<b>8.38 H-M</b>	<b>7.2 Meses</b>	<b>311.98 PM</b>	<b>\$2100.92 Udad Mon.</b>



### **Cálculo de costo de los medios técnicos.**

**CMT= Cdep + CE + CMTO** Donde:

- ♣ **Cdep:** Costo por depreciación.
- ♣ **CMTO:** Costo de mantenimiento de equipo
- ♣ **CE:** Costo por concepto de energía.

**CE= HTM x CEN x CKW** Donde:

- ♣ **HTM:** Horas de tiempo de máquina necesarias para el proyecto.
- ♣ **CEN:** Consumo total de energía
- ♣ **CKW:** Costo por Kwtas/horas (\$0.09 hasta 100 Kws \$ 0.20 de 101 a 300 Kws y \$ 0.30 mas de 300Kws)

**HTM= (Tdd x Kdd + Tip x Kip) x 152** Donde:

- ♣ **Tdd:** Tiempo promedio utilizado para el diseño y desarrollo (5 meses).
- ♣ **Kdd:** Coeficiente que indica el promedio de tiempo de diseño y desarrollo que se utilizó en la máquina (0.60)
- ♣ **Tip:** Tiempo utilizado para las pruebas de implementación (4 horas).
- ♣ **Kip:** Coeficiente que indica el % de tiempo de implementación utilizado en la máquina. (0.8)

$$\text{HTM} = (5 \times 0.60 + 4 \times 0.8) \times 152$$

$$\text{HTM} = (3.0 + 3.2) \times 152$$

$$\text{HTM} = 942.4 \text{ H//}$$

$$\text{CEN} = 0.608 \text{ Kw/h// (Estimado)}$$

$$\text{KW} = \text{HTM} \times \text{CEN}$$

$$\text{KW} = 942.4 \times 0.608$$





$$\text{KW} = 572.98 //$$

$$\text{CKW} = (100 \times 0.09) + (200 \times 0.20) + (572.98 \times 0.30)$$

$$\text{CE} = \$221 //$$

Resumiendo los cálculos de costo de energía tenemos:

Horas Tiempo Máq.	Cons. Total Energ.	KW	C. Energía
942.4 H	0.608 Kw/h	572.98	\$221

**Cálculo del costo de Materiales Técnicos:** El costo de utilización de los medios técnicos.

$$\text{CMT} = \$221$$

**Cálculo del Costo de Materiales:** En el cálculo de los costos de los materiales se consideró el 5 % de los costos de los medios técnicos.

$$\text{CMAT} = 0.05 \times \text{CMT} \text{ Donde:}$$

♣ CMT: Costo de los medios técnicos.

$$\text{CMAT} = 0.05 \times 221$$

$$\text{CMAT} = \$11.05$$

**Cálculo de Otros Gastos:** En otros gastos se tomó en cuenta un estimado de lo que se gastó por concepto de transporte y estadía para la realización del proyecto.

$$\text{OG: Se estima en } \$230$$



Después de realizados los cálculos correspondientes a los Costos Directos (CD), se obtienen los siguientes resultados.

$$\text{CD} = \text{CFT} + \text{CMT} + \text{CMAT} + \text{OG}$$

$$\text{CD} = 2100.92 + 221 + 11.05 + 230$$

$$\text{CD} = \$2562.97//$$

**Costo Total del Proyecto:** Para calcular el valor total del proyecto se utilizó la siguiente expresión:

$$\text{CTP} = \text{CD} + 0.1 \times \text{SB}$$

$$\text{CTP} = 2562.97 + 0.1 \times 15953.87$$

$$\text{CTP} = \$4158,36//$$

El costo de utilización de los medios técnicos y materiales quedaría en:

C. Med. Téc.	C. Mater.	Otros Gast.	C. Directos	C. Total Pr.
\$221	\$11.05	\$230	\$2562.97	\$4158,36

Los beneficios obtenidos por el uso de PENSOLOG son:

- ♣ Aumento de seguridad y fiabilidad de los datos.
- ♣ Aumento de la rapidez en la búsqueda y consulta de la información
- ♣ Emisión de Reportes seguros y confiables
- ♣ Ahorro de espacio físico para guardar la información.
- ♣ Ahorro de suministros de oficina
- ♣ Contribuye a la toma de decisiones oportunas
- ♣ Entrenamiento rápido y eficaz de alumnos y profesores.
- ♣ Contar con un medio de enseñanza más.



- ♣ Motivación eficiente en las clases.
- ♣ Propicia la formación, de alumnos y profesores, en cualquier lugar u hora

Comparando estos beneficios con el costo y más aún el hecho de que este producto se proveerá de forma gratuita, se determinó acometer la tarea de Diseñar y Desarrollar PENSOLOG y su Base de Datos. Los requerimientos utilizados para ello fueron:

### **RECURSOS HUMANOS:**

Se emplearon **3** personas durante un tiempo estimado de **5** meses.

Autor: Lic. Mayenny Linares Río

### **RECURSOS TECNOLÓGICOS:**

#### **HARDWARE:**

Componentes	Características
Procesador	Pentium IV 3.0 GHz
Memoria	256 RAM
Disco Duro	80 GB
Unidad de Respaldo	CD-ROM
Monitor	Aopen (1024 x 768)
Impresora	Inyección

#### **SOFTWARE**

- |                                    |                    |         |
|------------------------------------|--------------------|---------|
| ♣ Sistema Operativo<br>Windows XP. | ♣ ToolBook 8.5.    | ♣ Mash. |
|                                    | ♣ Adobe Photoshop. | ♣ Swish |

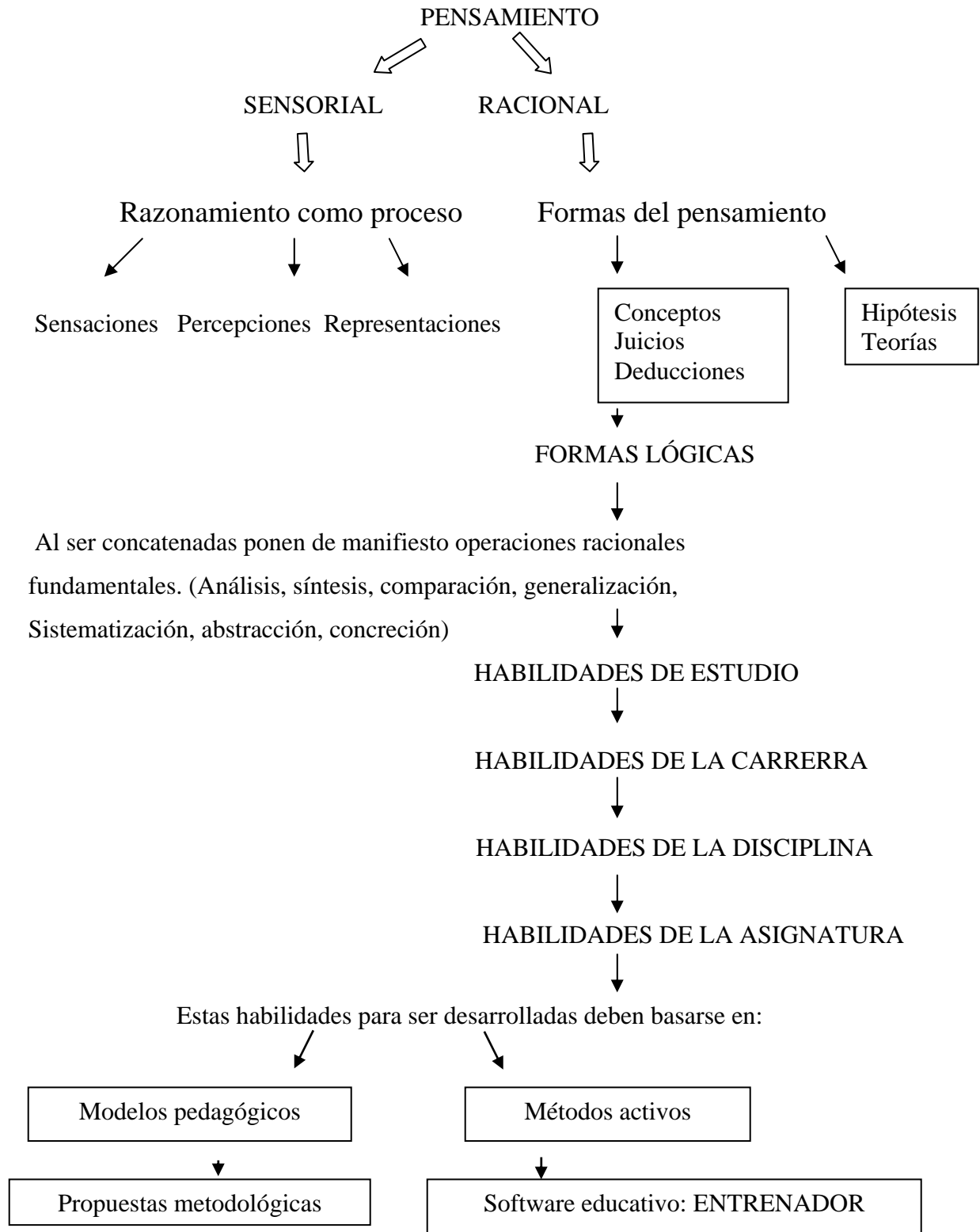


## **Conclusiones del capítulo II**

Al analizar el concepto de pensamiento llegamos a la relación existente entre el mismo y habilidades informáticas de ahí que para que este sea desarrollado correctamente, haya que partir del desarrollo de las habilidades mencionadas. (Ver esquema explicativo al final de estas conclusiones)

El software que se implementará deberá ser un entrenador por sus características, dentro de las que encontramos la de desarrollar habilidades, el diseñador decide qué entrenar, existe retroalimentación de los contenidos, se evalúa la participación del estudiante, sistema de refuerzo y la creación de expectativas.

El entrenador diseñado es objetivo, ya que es económico, ligero y está en función de las necesidades del cliente y el usuario.



**Esquema resumen que explica la relación entre pensamiento lógico y habilidades informáticas.**



### Capítulo III. Diseño e implementación de la propuesta de solución

#### Epígrafe 3.1. Análisis comparativo de la herramienta empleada para la implementación práctica del recurso.

##### Revisión del estado del arte de la tecnología.

En un inicio los programas debían ser realizados conociendo en bastante detalles los elementos del Hardware de la computadora en particular, esto hoy se puede seguir haciendo, pero es bastante tedioso y hay que conocer en un nivel básico la forma en que la computadora funciona, algunos detalles del hardware. Más tarde aparecieron los **lenguajes de alto nivel** o **superlenguajes**, que están más cerca de la notación algebraica natural que se utiliza para describir problemas.

Ejemplo de estos lenguajes pueden ser FORTRAN, ALGOL, COBOL, PL/I, Basic, Pascal, C, Ada, etc. Cada uno con sus peculiaridades, con sus características específicas. Después del surgimiento de la POO se ampliaron algunos de ellos y han dado lugar a ObjectPascal, C++ o algunos “puros” como Smalltalk.

El lenguaje Pascal fue diseñado en los años 70 por Niklaus Wirth, profesor el Instituto Politécnico Federal de Zurich. Este lenguaje, pensado en su origen para la enseñanza de la programación, ha sido adaptado a otros muchos propósitos, lo que permite emplear extensamente este lenguaje también para la programación práctica.

En años posteriores fue desarrollado el Object Pascal por Apple Computer con la asesoría del diseñador de Pascal, el propio Niklaus Wirth.

Estos lenguajes son muy utilizados para la creación de Multimedias, pero también se emplean los sistemas de autor (AuthorWare, **ToolBook**, Director, etc.)



Ante una estrategia de “elaboración masiva de aplicaciones multimedia educativas” en la que maestros y profesores se convierten en generadores de este tipo de medio de enseñanza, el empleo de lenguajes de programación de propósito general presenta inconvenientes significativos, el primero, es que tal enfoque exige un elevado dominio en materia de programación en la etapa de implementación; y el segundo, está vinculado con el tiempo de desarrollo que esto exige debido a que, por su carácter multipropósito, no son lenguajes especializados en tareas específicas sino que por el contrario, están concebidos para dar solución a la más diversa gama de problemas. En tal sentido la “no especialización” obliga al implementador a construir cada una de las “piezas” o “facilidades” que conformarán la aplicación desde un relativo bajo nivel con respecto a herramientas especializadas.

### **Las herramientas de autor**

Los sistemas de autor constituyen herramientas informáticas pensadas, en teoría, para desarrollar aplicaciones informáticas multimedia, concebidas para ser usadas por un maestro, profesor, un comunicador, publicista, guionista, sin que esto exija conocimientos especiales de programación. (Bou G, 1996).

Los sistemas de autor se caracterizan normalmente por estar asociados a una metáfora o modelo (libro, ventana, escritorio, organigramas, etc.) que simplifica la implantación de las ideas informáticas que se pretenden desarrollar, no sólo en términos de su interfaz multimedia, sino también permiten definir la interactividad con el sistema y en particular el flujo de navegación por la aplicación. Teniendo en cuenta la multiplicidad de tareas que tienen que ser resueltas al desarrollar una aplicación multimedia la mayoría de los sistemas de autor integran en su concepción un enfoque “multi-herramienta”, que garantiza el procesamiento de diferentes tipos de medios (gráfica, sonido, vídeo, etc.) mediante programas utilitarios propios, pero generalmente invocados desde la propia interface del entorno de trabajo del sistema.

Otra de las características inherentes a estos sistemas es la existencia de “macro-estructuras de alto nivel” que simplifican el logro de tareas en la fase de implementación. Estas estructuras se manifiestan en ocasiones en forma de conceptos abstractos (“clips”, “catálogos de widgets



.....*Análisis comparativo de la herramienta empleada para la implementación práctica del recurso.* (objetos pre-programados)”, “casting”, “behaviors”, diagramas de iconos y diálogos, etc.) que facilitan el logro de tareas que involucran a decenas de instrucciones de un lenguaje de propósito general.

La selección de una herramienta de autor dependerá fundamentalmente de dos factores: Las características particulares de la aplicación a desarrollar y la formación y experiencias del propio desarrollador. Además, es deseable que la metáfora del sistema facilite la implementación de la aplicación resultante y por otro lado es importante tener en cuenta las posibilidades del autor en materia de programación y la portabilidad de la aplicación hacia los sistemas operativos de los usuarios.

En la actualidad existen decenas de “sistemas de autor” que generalmente se agrupan en los siguientes paradigmas:

- a) Eje de los tiempos.
- b) Diagrama de flujo.
- c) Modelo Objeto.

Estos paradigmas generales adquieren concreción a través de diferentes metáforas como son: La metáfora de las diapositivas (Power Point), la metáfora del libro (ToolBook Instructor), la metáfora de organigramas (IconAuthor y Authorware), la metáfora de la producción filmica (Macromedia Director), entre otros.

Es usual encontrar en los sistemas de autor estructuras especializadas en la solución de tareas clásicas de la multimedia como son:

1. Carga y visualización de imágenes.
2. Uso de efectos de transición en la navegación o presentación de la información.
3. Ejecución y sincronización de archivos de sonido, vídeo y animaciones.
4. Definición de objetos visuales con funcionalidades específicas como son controles, objetos gráficos, contenedores y visualizadores de diferentes tipos de medios, ventanas, cuadros de diálogo, etcétera.





.....Análisis comparativo de la herramienta empleada para la implementación práctica del recurso.

5. Recepción de entradas del usuario y asignación de acciones en calidad de respuesta por parte del sistema.

De todas las herramientas mencionadas nos inclinamos a trabajar con el **TooolBook II Instructor**, ya que además de cumplir con todos los requerimientos antes mencionados, brinda la posibilidad de hacer exportaciones a otros lenguajes, como por ejemplo el PHP, denominados software libres y que pueden ser implementados en cualquier sistema operativo.

### **La herramienta de autor TooolBook II Instructor. Orígenes y proyección**

Asymetrix fue fundada en 1985 por Paul Allen, cofundador con Bill Gates de Microsoft. Su producto principal, ToolBook, tiene sus orígenes en un sistema diseñado para computadoras personales de tipo Macintosh denominado Hypercard que sale al mercado en agosto de 1987 e inmediatamente se convierte en el líder de los sistemas de autor del momento. En 1989 aparece la versión 1.0 de ToolBook que se distribuyó conjuntamente con la versión 3.0 de Windows. No fue hasta la versión 1.5 que aparece conjuntamente con la versión 3.1 de Windows que ToolBook gana atención especialmente en el sector educacional.

Su indiscutible similitud con el sistema autor Hypercard, el bajo costo relativo de las PC's con relación al Macintosh convirtió a ToolBook en un sistema autor con un atractivo natural basado en la metáfora del libro.

A partir de la versión 6.5, Asymetrix adopta la estrategia de generar dos productos muy similares pero con características específicas que permitan atender dos intenciones bien definidas:

**Instructor:** Herramienta de autor orientada a personal especializado en Informática y en particular al dominio de técnicas de programación.

**Assistant:** Sistema de autor orientado a personal docente, no especializado en Informática y en particular sin la exigencia del dominio de técnicas de programación.



En la actualidad ToolBook Instructor es una herramienta autor concebida para el desarrollo de aplicaciones educativas distribuibles en disquetes, CDROMs, redes locales y globales. Además de una amplia gama de software multimedia limitado prácticamente solo por la creatividad del autor, es posible crear cursos dinámicos e interactivos dotados de los más modernos recursos multimedia como sonido, animaciones, video, gráficos, etcétera.

Basado en tecnología hipermedia con ToolBook instructor es posible crear, personalizar y distribuir materiales didácticos para la educación a distancia, y en particular la enseñanza en línea.

Orientado hacia la enseñanza en línea, “Instructor” posee una serie de herramientas y estructuras predefinidas como son: sistemas de “asistentes”, “plantillas”, catálogos de objetos con funcionalidad predefinida (Widgets) que permiten desarrollar aplicaciones educativas a partir de estereotipos o plantillas preconcebidas, sin el empleo de un lenguaje de programación. Dentro de este concepto además existe un mecanismo de programación estrictamente visual denominado “Action Editor” (Editor de acciones) que permite personalizar el comportamiento de los objetos empleados, constituyendo esto una alternativa del lenguaje OpenScript (lenguaje de programación nativo de ToolBook). Es importante destacar que las aplicaciones diseñadas de esta manera pueden ser exportadas de forma plena a estándares de Internet como son HTML, Java y DHTML (HTML dinámico). En tal sentido, este enfoque ofrece una solución informática al problema denominado “cross platform” (plataforma cruzada)” que no es más que garantizar que una aplicación pueda correr en diferentes sistemas operativos (Windows, SO Mac, Unix, Linux, etcétera).

### **¿Qué aplicaciones se pueden desarrollar con ToolBook Instructor?**

Con la herramienta autor ToolBook Instructor además de aplicaciones educativas basadas en hipermedia, también llamados “hiperentornos educativos”, se pueden hacer prácticamente cualquiera de las aplicaciones informáticas que usualmente se realizan con lenguajes de propósito general.



La existencia de un poderoso entorno de programación basado en el lenguaje orientado a objetos y dirigido por eventos denominado **OpenScript** dota al sistema de la flexibilidad necesaria para desarrollar la más amplia gama de aplicaciones bajo el sistema operativo **Windows**. (Windows 9x, Windows ME, Windows NT, Windows 2000).

La siguiente enumeración presenta los diferentes tipos de aplicaciones factibles de realizar con ToolBook. Esta enumeración puede ser ampliada de acuerdo con la creatividad de los desarrolladores.

- a) Presentaciones electrónicas.
- b) Libros electrónicos basados en hipermedia.
- c) Bases de datos.
- d) Tutoriales.
- e) Entrenadores.
- f) Juegos instructivos.
- g) Simuladores.
- h) Utilitarios.
- i) Enciclopedias digitales.
- j) Catálogos.
- k) Glosarios especializados.
- l) Sitios Web de diversa naturaleza, en particular educativos.
- m) Materiales didácticos para la educación a distancia.

Tollbook cuenta con varios paneles, que contienen un grupo de objetos pre-progamados y que pueden ser transformados a conveniencia del autor, dentro de ellos los más utilizados fueron:

A cada uno de estos objetos se le asignaron propiedades propias tales como:

- ♣ Transparencia
- ♣ Color
- ♣ Fuente



.....*Análisis comparativo de la herramienta empleada para la implementación práctica del recurso.*

- ♣ Alineación
- ♣ Contenido
- ♣ Visibilidad

Para la inserción de imágenes de fondo y los botones de cada ventana, fueron creados logotipos, en photoshop, para cada asignatura, en lo que tuvieron participación los estudiantes.

Se crearon formularios para cada asignatura en los que se diferencian sus colores e imágenes, pero todos contienen el mismo formato, para no perder la unicidad de la interfaz que se emplea.

El control del aprendizaje se lleva a cabo a través de una base de datos creada en la misma aplicación, la cual está contenida en un mismo libro y basada en la generación de páginas que funcionan como formularios.



## **Epígrafe 2. Descripción de las herramientas utilizadas para la implementación del software.**

En la elaboración del software propuesto, hubo de utilizarse herramientas auxiliares que facilitaran el diseño gráfico del mismo, la creación de animaciones y la de las mascotas.

Para el diseño gráfico se utilizó el Adobe Photoshop, con el completo juego de herramientas Web, de retoque, de pintura y de dibujo, el cual ayuda a completar eficazmente cualquier tarea de edición de imágenes. Además, con funciones como la paleta Historia y los efectos de capa editables, se puede experimentar con total libertad sin tener que sacrificar la eficacia.

La mayoría de las herramientas cuentan con opciones que aparecen en la barra de opciones de la herramienta. La barra de opciones es sensible al contexto y cambia según la herramienta que se seleccione. Algunos ajustes de la barra de opciones son comunes a las distintas herramientas (por ejemplo, los modos de pintura y la opacidad) y otros, específicos de una herramienta (por ejemplo, la opción Borrado automático de la herramienta Lápiz).

Para el tratamiento de imágenes se hicieron transformaciones a través de las opciones de fusión como se muestra en la figura 1 que aparecen en el Anexo XXI.

Se establecen sus propiedades de interfaz como se muestra en la figura 2 que aparecen en el Anexo XXI.

Dando un acabado final a través de los diferentes ajustes de imágenes como se muestra en la figura 3 que aparecen en el Anexo XXI.

Esta aplicación nos permite la aplicación de filtros y propiedades que hacen posible optimizar la calidad de la imagen tratada.



## **2. Para la implementación de las mascotas el Mash.**

El Mash es una herramienta case, que a partir de la configuración de los elementos que nos brinda, podemos obtener fácilmente la animación de mascotas. Ella cuenta con selección de lenguaje, de la mascota y sus acciones específicas, así como la forma en que aparecerán los mensajes. Veamos ejemplos:

Lo primero que debemos hacer es seleccionar la mascota que queremos animar como se muestra en la figura 4 que aparecen en el Anexo XXI.

Después seleccionamos la acción a ejecutar como se muestra en la figura 5 que aparecen en el Anexo XXI.

Si queremos que diga un texto en específico debemos configurar el lenguaje y los formatos de los balones en que muestran los mismos como se muestra en la figura 6 y 7 que aparece en el Anexo XXI.

Así vamos agregando acciones y textos según nuestra conveniencia para luego guardar como \*.exe.

## **3. Para clic de presentación el Swish.**

SWiSH es fácil de usar, permite producir complejas animaciones con texto, imágenes, gráficos y sonido en muy poco tiempo. SWiSH tiene más de 150 efectos como Explosión, Vórtice, Giro 3D, Serpiente y muchos más. SWiSH tiene herramientas para crear líneas, rectángulos, elipses, curvas de Bezier, movimientos, Sprites, botones animados, y todo en un sencillo interfaz.

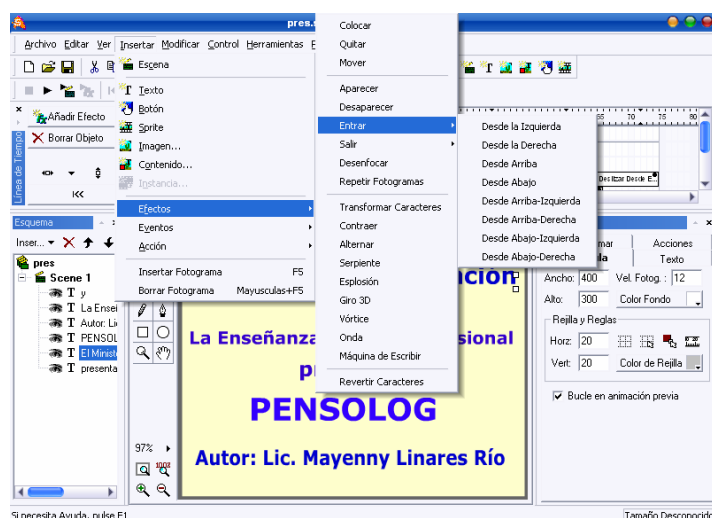
La interfaz de usuario de Swish 2 consiste en un conjunto de "paneles". Cada panel muestra una vista diferente de la película Swish o propiedades del objeto seleccionado. Los paneles incluyen: Diseño, Esquema, Línea de Tiempo, Película, Escena, Forma, Texto, Imagen, Transformación, Mezclador, Acciones, Contenidos y Exportación. Los paneles pueden estar flotando en la pantalla o endosados a barras de herramientas. Tanto las barras de herramientas



.....Descripción de las herramientas utilizadas.  
como los menús son configurables por el usuario, y se pueden asignar atajos de teclado a cualquier comando de menú.

En esta aplicación las herramientas más utilizadas fueron la inserción de textos e imágenes, así como la transformación del fondo de la escena como se muestra en la figura 8 del Anexo XXI.

Además se le asignaron efecto de entrada y salida a los textos y a las imágenes, siguiente el siguiente accionar:



Y se finalizó insertando sonido a la película y exportándola con extensión \*. avi, para que pudiera ser aceptada por el TollBook.



### Epígrafe 3. Implementación práctica.

Dentro de los tipos de software educativos mencionados, se ha optado por el Entrenador, para el desarrollo del recurso, pues como ya se había puntualizado este posibilita el desarrollo de habilidades en el usuario con un alto grado de independencia. Además brinda las facilidades de poder mostrar los ejercicios de forma aleatoria, controlar su aprendizaje y retroalimentar los contenidos. Dicho sistema contendrá rasgos de tutorial para que el estudiante pueda acceder a un mayor grupo de información que deba ser utilizado como base para su preparación.

El sistema recomendado (PENSOLOG) contiene un grupo de interfaces que posibilitan la navegación por el mismo y cuyo color predominante es el azul, por ser este el color preferido de la gran mayoría de los alumnos, el resto de los colores presentados en las pantallas de cada una de las asignaturas está acorde con el que se muestra en los logotipos del sistema operativo Windows; en aquellos casos en que la aplicación no exista se le asignó el color decidido por los estudiantes y profesores.

El diseño de las imágenes de cada interfaz fue determinado por los propios estudiantes y los ejercicios elaborados sobre la base de la preparación de los profesores.

A continuación se describen las mismas después de haber sido implementadas.

#### **Pantalla: Identificación.**

En esta pantalla el usuario selecciona su condición y a partir de ahí se le va orientando el camino en función del objetivo del software asociado a él.

Esta selección se realiza haciendo clic sobre el tipo de usuario que no es más que un botón con propiedades de etiqueta, transparencia y fuente verdana 12, en negrita. A este se le asocia un lenguaje de código compartido, como se muestra en la figura 1, para cada botón y en sus propiedades (ver figura 2 y 3 en el Anexo XXII) se le asigna los valores miLibro y miPag en los cuales se pone el camino al que se dirigirá.

```
to handle buttonClick
System CaminoEXE
Libro = CaminoExE & "Sonido\Click.wav"
get playSound (Libro)
miPag = miPag of self
miLibro = miLibro of self
send navega miLibro, miPag
end buttonClick
```

.....

**Figura 1. Lenguaje de código para la navegación.**





### **Pantalla: Autenticación.**

En esta pantalla el estudiante llena el formulario con los datos que se les pide y al dar clic en el botón comenzar, estos datos son transferido a un fichero \*.txt (figura 5) y da paso a la pantalla principal. Esta entrada de datos está condicionada poner solo letras (figura 4).

```
to handle enterfield
  system estado0, estado1
  estado0 = keyState(keyNumLock)
  estado1 = keyState(keyCapital)
  if estado0 = "Down"
    -- request "Tienes el estado numérico encendido"
  end if

  if estado1 = "down"
    -- request "Tienes el estado mayúscula encendido"
  end if
end
-----
```

**Figura 4. Código para entrada de nombre.**

```
to handle ButtonClick
-----

system FormaTrab
system ANombre[4],AGrupo[4],AMiTiempo[4],AGrado[4],BNombre[4]
System MiTiempo
systemFormat = seconds
MiTiempo = systime

AMiTiempo[1] = MiTiempo+1
AMiTiempo[2] = MiTiempo+2
AMiTiempo[3] = MiTiempo+3
AMiTiempo[4] = MiTiempo+4
FormaTrab = DameFormaTrab()
.....
```

**Figura 5. Código para validación de entrada y almacenamiento en fichero \*.txt.**



### **Pantalla: Principal**

En esta pantalla el estudiante podrá seleccionar el tema en el que quiere entrenarse, puede escuchar un instrumental como fondo musical (figura 8) o acceder a la ayuda, en la cual aparece una mascota explicando las funcionalidades de la pantalla, así como ir a la página principal. Todo esto a través de botones con la misma tipología que los de la pantalla de autenticación.

```
-- SharedScript "ASYM_RunApp"

to get ASYMA_EditWidgetProps
  oldFile = my app
  cap = ASYM_GetString("capOpenFile","Choose a file")

  defFile = "*.*)"
  defPath = Null
```

**Figura 8. Código para ejecutar aplicación.**

### **Pantalla: Ejercicios**

En esta pantalla el estudiante podrá dar respuesta a los ejercicios que se les presentan de forma aleatoria, puede escuchar un instrumental como fondo musical o acceder a la ayuda, en la cual aparece una mascota explicando las funcionalidades de la pantalla, ir a la página principal, ver imágenes, textos o glosario técnico asociados al tema que se trata. Además puede visualizar su ubicación específica dentro del recurso en esos momento, el tiempo que demoró en resolver el ejercicio y la puntuación alcanzada. En este módulo existen varias topologías de ejercicios basados en módulos de programación prediseñados por el departamento de creación de software de nuestra provincia.

### **Pantalla: Ayuda**

En esta pantalla el estudiante podrá visualizar los contenidos relacionados con el tema en el que está trabajando, puede escuchar un instrumental como fondo musical o ir a la pantalla principal e imprimir estos textos. (Igual que en el resto de las pantallas y como se muestra en las figuras 5, 6 y 7 del Anexo XXII)

### **Pantalla: Glosario Técnico.**

En esta pantalla el estudiante podrá visualizar los conceptos relacionados con el tema en el que está trabajando, puede escuchar un instrumental como fondo musical o ir a la pantalla principal e imprimir estos textos. (Igual que en la pantalla ayuda)



**Pantalla: Imágenes.**

En esta pantalla el estudiante podrá visualizar las imágenes (botones) relacionadas con el tema en el que está trabajando, de la forma en que se ofrece o agrandada (ver código en la figura 8), puede escuchar un instrumental como fondo musical o ir a la pantalla principal e imprimir estos textos.

**Pantalla: videos.**

En esta pantalla el estudiante podrá visualizar los videos relacionados con el tema en el que está trabajando, puede escuchar un instrumental como fondo musical, ir a la pantalla principal u obtener ayuda sobre su funcionamiento. (Ver figuras 9 del Anexo XXII)

**Pantalla: Base de datos.**

En esta pantalla el profesor podrá visualizar los datos relacionados con el tema en el que está trabajando, navegar por toda la información, así como gestionarla. (Ver sus códigos en las figuras de la 10 a la 13 del Anexo XXII)

**Pantalla: Selección.**

En esta pantalla el profesor podrá seleccionar el tipo de tarea que desea realizar con el sistema, cerrar o minimizar la ventana. (Igual que en la de autenticación)

**Pantalla: Gestión de ejercicios.**

En esta pantalla el profesor agregará, eliminará o modificará el (los) ejercicio (s) que desee. Puede además escuchar un instrumental como fondo e ir a la pantalla de selección. (Igual que en el resto de las pantallas)

Para la puesta en práctica de este recurso, el centro debe adoptar condiciones favorables y de hecho ya existen. Constamos con cinco laboratorios con tecnología moderna, cada uno consta con 16 máquinas HANEL P4, conectadas en redes controladas desde un servidor. Los estudiantes están sentados 2 por cada máquina durante el proceso docente.

Todos los laboratorios están climatizados con dos consolas a 16 grados centígrados o a la opción de ventilador. Encontramos un técnico en cada laboratorio como máximo responsable de la documentación y equipamiento del laboratorio. Tendrán posibilidad de acceder al Software a través de las clases de la unidad de estudio número 1 de la asignatura de Proyectos Informáticos, estudios independientes dirigidos por el maestro, y en los tiempos de máquina.



Las máquinas están instaladas con Service Pack 2 y todo el software necesarios para poder acceder de manera rápida y segura por el servicio FTP a dicho sistema. Tenemos como protector de los virus al antivirus Kaspersky donde está predeterminado actualizarse automáticamente cada dos días en cada máquina de los laboratorios y de los departamentos para una mayor seguridad de protección a los medios tecnológicos puestos por nuestra Revolución.

La explotación de este entrenador puede llevarse a cabo dentro y fuera de clases y su uso puede ser mediado o no por el profesor. (Ver ejemplo de una de las vías de utilización en Anexo XV y su manual de usuario en el anexo XVI)



### **Conclusiones del capítulo III.**

Para la implementación del sistema se hace necesaria la utilización de una herramienta que brinde facilidades en su empleo, para lograr optimizar el trabajo en cuanto a tiempo, complejidad y costo.

De todas las herramientas existentes nos inclinamos a trabajar con el **TooolBook II Instructor**, ya que con él podemos crear una amplia gama de software multimedia limitado prácticamente solo por la creatividad del autor, es posible crear cursos dinámicos e interactivos dotados de los más modernos recursos multimedia como sonido, animaciones, video, gráficos, etcétera. Además de cumplir con todos los requerimientos antes mencionados, brinda la posibilidad de hacer exportaciones a otros lenguajes, como por ejemplo el PHP, denominados software libres y que pueden ser implementados en cualquier sistema operativo.



## Conclusiones

Con el desarrollo de este trabajo hemos podido elaborar un software entrenador **PENSOLOG** que contribuya al desarrollo del pensamiento lógico de los alumnos con apoyo de un sistema de tareas, en la asignatura Proyectos Informáticos, el cual cumple con los requisitos de diseño establecidos y con las necesidades y expectativas del cliente. Además el centro en el que será explotado cuenta con las condiciones materiales y humanas necesarias para su uso.

Se logró crear una base de datos, acumulativa de la información pertinente, para el desarrollo del software, a través de la cual los profesores podrán, visualizar y administrar los datos de los alumnos y los ejercicios.

Se elaboraron un grupo de actividades metodológicas que incluyen las tareas de apoyo al software PENSOLOG, en la asignatura Proyectos Informáticos, las cuales propician el entendimiento del recurso por parte de los profesores y la preparación de los mismos para la correcta explotación del sistema.



## **Recomendaciones**

Implementar el módulo de ejercicios en el resto de las asignaturas.

Implementar el sistema en los niveles siguientes, de la carrera.

Generalizar el sistema en las asignaturas de formación no técnica.

Implantar el sistema en los Politécnicos de informática del país.



## **Referencias Bibliográficas**

1. Serrano Gómez A, Díaz Hernández R, M Rodríguez Lamas R, González Chong O, Pigueiras D, Martín Viera L, Aguila Iglesias M, Del P. Alea Díaz M Smirnov, Leontiev y otros. Psicología. Imprenta nacional de cuba; 1961
2. Guadarrama González Pablo, Suárez Gómez Carmen. Filosofía y Sociedad. Tomo II. Editora "Félix Varela"; 2001.
3. Castellanos Simona D. La comprensión de los procesos del aprendizaje: apuntes para un marco conceptual. 2da. Versión, 1999 febrero.
4. Konstantinov F, otros. Fundamentos de la Filosofía Marxista – Leninista. Materialismo Dialéctico. Tomo II. Editorial Pueblo y Educación; 1986.
5. De Armas Ramírez N, Lorences González J, Perdomo Vázquez JM. Caracterización y diseño de los resultados científicos como aportes de la investigación educativa. Universidad Pedagógica "Félix Varela"
6. Comisión de Planes y Programas formada al efecto e Integrada por compañeros de: UCI, FORDES, Correos de Cuba por el MIC, CUJAE por el MES, ISPETP, ISPEJV, DFPPP, ETP por el MINED. Plan de Estudio y Programas para los nuevos Politécnicos de Informática. 30 de Junio del 2005
7. Psicología General. Manual Didáctico para los Institutos de Pedagogía. Redactado por el profesor A. Petrovki. Tercera parte: Procesos cognitivos del individuo. Editorial Progreso; 1980. Capítulo XII: El pensamiento, p. 292.
8. Miqueo Domínguez J. Entrenador matemático para la geometría plana que se imparte en la enseñanza general media. [Tesis de Maestría [. Cuba, Pinar del Río: Instituto Superior Pedagógico:"Rafael María de Mendive"; 1998
9. Prives M, Lisenkov N, Bushkovich V. Anatomía Humana. Tomo II. Editorial MIR; 1975
10. Rodríguez Piñeiro Alberto J. "Entrenador para la Geometría Descriptiva". [Tesis de Maestría [. Cuba, La Habana; 1997.
11. Documento del MINED. Consideraciones sobre la estructura interna y el algoritmo de algunas habilidades.





12. Nocedo de León I, Castellanos Simona B, García Batista G, Advine Fernández F, González Dosil C, Gort Sánchez M, Ruiz Aguilera A, Minujín Zmud AF, Valera Alfonso O. Metodología de la investigación educacional. Segunda parte. Primera reimpresión. Cuba: Editorial Pueblo y Educación ; 2002
13. Texto del discurso pronunciado por el Presidente Fidel Castro en el acto de inicio del curso escolar 2003-2004, efectuado el 8 de septiembre en la Plaza de la revolución José Martí.
14. Ramos Rivero P. De Los “Teleniños” A Los “Ciberniños”. Mutaciones En El Entorno Hipermedial. La Habana
15. Zaldívar Carrillo ME. , Sosa Oliva Y. El Desarrollo Del Pensamiento De Los Estudiantes A Través De La Enseñanza. Cuba. Revista Iberoamericana de Educación (ISSN: 1681-5653). Instituto Superior Pedagógico “José de la Luz y Caballero”.
16. Trujillo Sainz JA. Introducción a la Informática Educativa. Universidad de Pinar del Río Hermanos Saíz, Instituto Superior Politécnico José A. Echeverría. República de Cuba; 2000
17. Barrio M. Universalización de la Universidad. Una nueva forma de aportar felicidad. Diario de la Juventud cubana. 2004, marzo, 28.

## **Anexo I. Guía de entrevista a profesores.**

Objetivo: Determinar los factores que hacen posible el aprendizaje insuficiente de los alumnos.

Categorías:

- ♣ Conocimiento del perfil del egresado.
- ♣ Dominio de las habilidades.
- ♣ Preparación de los profesores
- ♣ Aprendizaje de los alumnos.

Cuestionario:

1. ¿Cuáles son las causas que motivan el hecho de que los alumnos no aprendan lo suficiente?
2. ¿Qué habilidades debe formar en sus alumnos para que egresen como informáticos capaces?
3. ¿Qué deben saber hacer sus alumnos en su asignatura?
4. ¿Qué acciones lleva usted a cabo para desarrollar la habilidad de modelar?

## **Anexo II. Guía de encuesta aplicada a profesores.**

Objetivo: Determinar los factores que hacen posible el aprendizaje insuficiente de los alumnos.

Categorías:

- ♣ Motivación de los alumnos.
- ♣ Dominio de las habilidades.
- ♣ Aprendizaje de los alumnos.

Cuestionario:

1. Mencione en orden jerárquico, cuáles son las causas, que a su entender, hacen posible el aprendizaje insuficiente de los alumnos.
2. ¿Qué parte de la metodología que aplica en sus clases, no le es de su agrado? ¿Por qué? Cite algunos ejemplos.
3. Proponga algunas variantes que le ayudarían a mejorar algunas de las deficiencias citadas anteriormente. Ejemplifique las propuestas.

### **Anexo III. Guía de entrevista aplicada a los alumnos.**

Objetivo: Valorar la motivación que tienen los estudiantes por el estudio de la Informática y el conocimiento sobre lo que deben saber hacer como futuros técnicos medios en esta especialidad.

Categorías:

- ♣ Motivación de los alumnos.
- ♣ Conocimiento del papel del egresado.
- ♣ Aprendizaje de los alumnos.

Cuestionario:

1. ¿Te motivan las clases de las asignaturas técnicas?
2. ¿Cuál asignatura te motiva menos? ¿Por qué?
3. ¿Conoces lo que debes saber hacer como informático cuando culmines tus estudios en la escuela?

#### **Anexo IV. Guía de encuesta aplicada a los estudiantes.**

Objetivo: Conocer los criterios, de los alumnos, acerca de la forma en que se imparten las clases de las asignaturas técnicas y de cómo le gustaría que se impartieran.

Categorías:

- ♣ Motivación de los alumnos.
- ♣ Conocimiento del perfil del egresado

Cuestionario:

1. ¿Te motivan las clases de las asignaturas técnicas?
2. ¿Cómo las mejorarías?
3. ¿Qué te gustaría hacer para entretenerte?
4. ¿Qué debes saber hacer como informático cuando te gradúes?

## Anexo V. Guía de observación a clases.

Objetivo: Valorar el aprendizaje de los alumnos.

Categorías:

- ♣ Motivación de los alumnos.
- ♣ Dominio de las habilidades.
- ♣ Aprendizaje de los alumnos.

Cuestionario:	Poco	Medio	Muy
1. Motivados	_____	_____	_____
• Permanencia frente a la PC.	_____	_____	_____
• Trabajo continuo en la PC.	_____	_____	_____
• Búsqueda de alternativas para lograr resolver los ejercicios	_____	_____	_____
2. Dominio de las habilidades.	_____	_____	_____
• Análisis del problema propuesto.	_____	_____	_____
• Búsqueda de la vía de solución.	_____	_____	_____
• Aplicación de los contenidos estudiados.	_____	_____	_____
• Uso correcto de las herramientas de trabajo.	_____	_____	_____

## **Anexo VI. Guía de revisión de documentos.**

Objetivo:

- Valorar el aprendizaje alcanzado por los alumnos hasta el momento.

Categorías a evaluar:

- ♣ Aprendizaje de los alumnos.
- ♣ Dominio de las habilidades informáticas.

## **Anexo VII. Guía de revisión de documentos (Plan de estudio)**

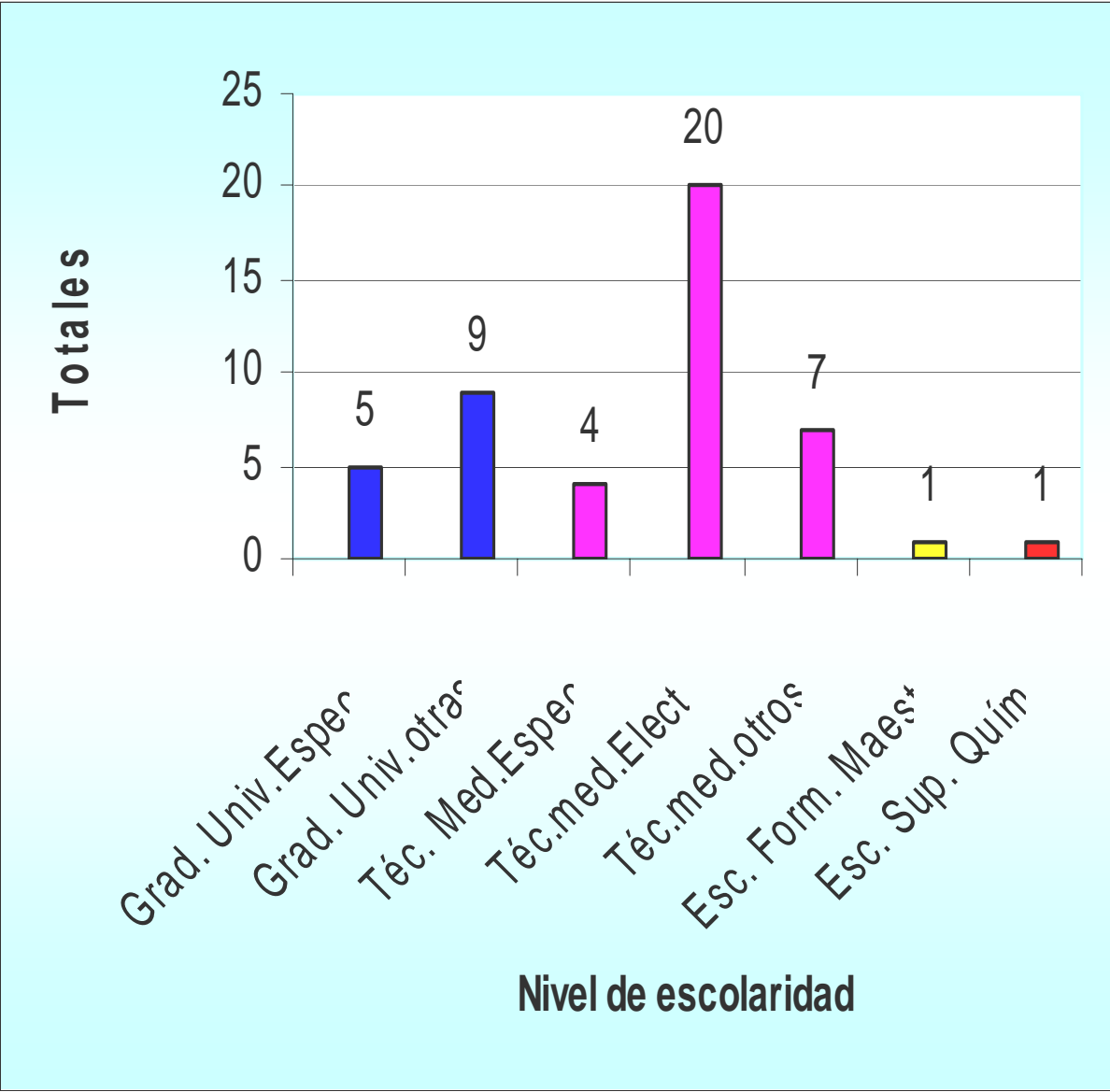
**Objetivo:** Valoración crítica de la fundamentación e instrumentación del aprendizaje de la Informática.

### **Categorías a evaluar:**

- ♣ Identificación de las habilidades informáticas.
- ♣ Derivación gradual de las habilidades.
- ♣ Enfoque de sistema.
- ♣ Relación con los conocimientos y valores.
- ♣ Metodología de trabajo.



**Anexo VIII. Gráfico que representa la cantidad de profesores de la especialidad, según su nivel de escolaridad.**



## **Anexo IX. Programa para la puesta en práctica de la propuesta.**

### **Propuesta metodológica.**

**Fundamentación:** Para desarrollar el pensamiento lógico de los alumnos se hace necesario instruir a los profesores en las vías a seguir para lograrlo (las habilidades que lo posibilitan y cómo lograr dar tratamiento a estas habilidades de forma tal que se desarrollen a la par de este tipo de pensamiento).

#### **Fase preparatoria:**

##### **Actividades a desarrollar en esta fase:**

1. Valoración crítica de los programas de estudio.
2. Análisis de las habilidades que tributan al desarrollo del pensamiento lógico. (Anexo XI)
3. Análisis de la estructura interna de las habilidades definidas (Anexo XII)
4. Establecer el carácter de sistema de estas habilidades y su relación con los valores. ( Anexo XIII)
5. Estudio de los temas que se imparten en las asignaturas.
6. Análisis de los ejercicios que se proponen por temas.
7. Confección de una propuesta de ejercicios por asignaturas en los que se visualice la búsqueda de vías de soluciones como elemento rector en el desarrollo del pensamiento lógico.

#### **Fase ejecutoria:**

**Sistema de actividades metodológicas que propiciarán el tratamiento de las habilidades para desarrollar el pensamiento lógico.**

##### **Actividades a desarrollar en esta etapa.**

1. Impartir una actividad metodológica a los profesores con el objetivo de demostrar las formas de relacionar el desarrollo del pensamiento lógico con el desarrollo de habilidades informáticas, a través de la búsqueda de vías de solución de situaciones problemáticas. (Ver Anexo XIV)
2. Orientar, a los profesores, la confección de ejercicios de este tipo donde se muestre vías de solución.
3. Debate de las propuesta de ejercicios creadas.

- Implementación de estas propuestas a un entrenador, para ser utilizado en clases como medio de enseñanza que posibilita el tratamiento de este tipo de ejercicios.
- Demostrar a los profesores el correcto uso y explotación del software que se propone, a través de una clase (Ver Anexo XV).

### **Programa para la preparación de los profesores.**

#### **OBJETIVOS GENERALES.**

- Capacitar a los profesores en el tratamiento de las habilidades informáticas.
- Instruir a los profesores en las formas de tratamiento de los ejercicios en la clase, para elevar el desarrollo del pensamiento lógico de los alumnos.

#### **PLAN TEMÁTICO Y DISTRIBUCIÓN DE HORAS.**

TEMÁTICA	HORAS		
	TOTAL	PRÁCTICA	EVALUACIÓN
Relación entre habilidades y pensamiento lógico.	1	1	
Carácter de sistema de las habilidades y su relación con los valores	1	1	
Confección de una propuesta de ejercicios por asignaturas	1	1	
Debate de las propuesta de ejercicios creadas.	1	1	1
Taller metodológico a los profesores sobre el correcto uso y explotación del software que se propone.	1	1	
Examen Final			2

#### **Invariantes del conocimiento.**

- Análisis de las habilidades que tributan al desarrollo del pensamiento lógico.
- Análisis de los ejercicios que se proponen por temas.
- Confección de una propuesta de ejercicios por asignaturas en los que se visualice la búsqueda de vías de soluciones como elemento rector en el desarrollo del pensamiento lógico.

#### **Habilidades a desarrollar**

- Interpretar.

- 1.1. Analizar el objeto.
  - 1.1.1. Determinar límites del objeto.
  - 1.1.2. Determinar criterios de descomposición
  - 1.1.3. Determinar partes del todo
  - 1.1.4. Determinar cada parte delimitada.
- 1.2. Relacionar las partes del objeto.
  - 1.2.1. Analizar de manera independiente los objetos a relacionar.
  - 1.2.2. Determinar criterios de relación
  - 1.2.3. Determinar nexos entre los objetos.
  - 1.2.4. Determinar nexos inversos.
  - 1.2.5. Elaborar conclusiones.
- 1.3. Encontrar la lógica de la relación encontrada.
- 1.4. Elaborar conclusiones acerca de los elementos.

## 2. Modelar.

- 2.1. Análisis del objeto.
  - 2.1.1. Determinar los límites del objeto.
  - 2.1.2. Determinar criterios de descomposición.
  - 2.1.3. Determinar las partes del todo.
  - 2.1.4. determinar cada parte determinada.
- 2.2. Interpretar cada parte del objeto.
  - 2.2.1. Analizar el objeto.
  - 2.2.2. Relacionar las partes del objeto.
  - 2.2.3. Encontrar la lógica de la relación prevista.
  - 2.2.4. Elaborar conclusiones acerca de los elementos.
- 2.3. Diseñar el objeto.
  - 2.3.1. Caracterizar el objeto.
  - 2.3.2. Identificar las principales características.
  - 2.3.3. Esbozar según sus principales características.
  - 2.3.4. Determinar los recursos y procedimientos informáticos a emplear para su diseño.
  - 2.3.5. Elaborar el objeto.
- 2.4. Aplicar procedimientos informáticos para obtener el modelo.

- 2.4.1. Determinar los objetos.
- 2.4.2. Confirmar el dominio de conocimientos que se pretende aplicar al objeto.
- 2.4.3. Caracterizar el objeto donde se quiere aplicar.
- 2.4.4. Interrelacionar los conocimientos con el objeto.
- 2.4.5. Elaborar conclusiones.
- 2.5. evaluar el objeto.
- 2.6. Aplicar transformaciones en caso de ser necesario.

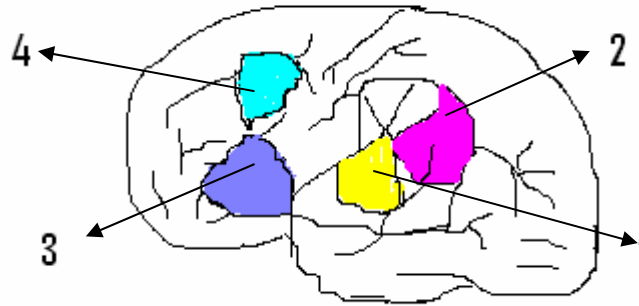
### **Sistema de conocimientos**

- Análisis de las habilidades que tributan al desarrollo del pensamiento lógico: definición de pensamiento lógico, definición de habilidades y de habilidades informáticas, relación del pensamiento lógico con las habilidades informáticas.
- Análisis de los ejercicios que se proponen por temas: Características de los ejercicios, características que deben tener los ejercicios para lograr desarrollar el pensamiento lógico, presentación de ejemplos.
- Confección de una propuesta de ejercicios por asignaturas en los que se visualice la búsqueda de vías de soluciones como elemento rector en el desarrollo del pensamiento lógico: Determinar las habilidades a desarrollar por cada asignatura, relación habilidad-contenido, creación de ejercicios teniendo en cuenta la habilidad a desarrollar, el contenido a tratar y sus características.

### **Evaluación.**

La evaluación final consistirá en un debate entre los profesores de cada departamento sobre la propuesta de ejercicios planteada. En él se analizarán el mayor número de ejemplos posibles, siendo los mismos profesores los que los expongan. Se deberá tener en cuenta la utilización de los medios tecnológicos en la propuesta.

**Anexo X. Hemisferio cerebral izquierdo.**



1. Centro acústico.
2. Centro óptico del lenguaje.
3. Centro motor del lenguaje oral.
4. Centro motor del lenguaje escrito.

**ANEXO XI. Habilidades que deben desarrollar los estudiantes de los Politécnicos de Informática. Propuesta de una adecuada derivación gradual.**

<b>HABILIDADES</b>		
<b>GENERALES</b>	<b>ESPECIFICAS</b>	
Explotar PC	Preparar	Instalar y configurar computadoras y sus periféricos. Instalar y actualizar aplicaciones informáticas.
	Operar	Utilizar buscadores y navegadores de páginas Web y aplicaciones informáticas. Ejecutar cambios para mejorar la explotación del sistema y las aplicaciones.
	Mantener	Prestar ayuda especializada para el uso correcto del equipo y las aplicaciones a los usuarios. Ejecutar el reglamento de seguridad informática. Diagnosticar el origen de las fallas en las computadoras y sus periféricos.
	Controlar	Propone y coordina cambios para mejorar la explotación del sistema y las aplicaciones
Programar	Interpretar	Interpretar diseños de ingeniería de software. Diseñar bases de datos.
	Implementar	Seleccionar herramientas y lenguajes apropiados cuando no se especifique. Codificar algoritmos y aplicaciones (generales, Web y Bases de Datos) Comentar y documentar el código.
	Depurar	Utilizar juegos de datos necesarios y herramientas para probar el software. Diagnosticar el origen de las fallas en el software. Poner a punto el software.
	Mantener	Proponer, coordinar y ejecutar cambios para mejorar la funcionalidad del software.

## **Anexo XII. Estructura interna de las habilidades**

### **1. Algoritmizar.**

#### **1.1. Determinar el modelo a seguir**

#### **1.2. Relacionar con los contenidos matemáticos que se conocen.**

1.2.1. Analizar de manera independiente los objetos a relacionar.

1.2.2. Determinar criterios de relación.

1.2.3. Determinar nexos entre los objetos.

1.2.4. Determinar nexos inversos.

#### **1.3. Definir la secuencia lógica de acciones a seguir en la solución.**

1.3.1. Determinar las características esenciales que distinguen el  
objeto.

1.3.2. Enunciar de forma sintética los rasgos esenciales del objeto.

#### **1.4. Convertir la secuencia de acciones lógicas en lenguaje de pseudocódigo.**

### **2. Codificar.**

#### **2.1. Interpretar los datos.**

#### **2.2. Relacionar los datos con los códigos asociados a él.**

#### **2.3. Convertir del lenguaje de pseudocódigo al de código según la aplicación empleada.**



## 2. Modelar.

### 1.1. Análisis del objeto.

1.1.1. Determinar los límites del objeto.

1.1.2. Determinar criterios de descomposición.

1.1.3. Determinar las partes del todo.

1.1.4. determinar cada parte determinada.

### 1.2. Interpretar cada parte del objeto.

1.2.1. Analizar el objeto.

1.2.2. Relacionar las partes del objeto.

1.2.3. Encontrar la lógica de la relación prevista.

1.2.4. Elaborar conclusiones acerca de los elementos.

### 1.3. Diseñar el objeto.

1.3.1. Caracterizar el objeto.

1.3.2. Identificar las principales características.

1.3.3. Esbozar según sus principales características.

1.3.4. Determinar los recursos y procedimientos informáticos a emplear para su diseño.

1.3.5. Elaborar el objeto.

### 1.4. Aplicar procedimientos informáticos para obtener el modelo.

1.4.1. Determinar los objetos.

1.4.2. Confirmar el dominio de conocimientos que se pretende aplicar al objeto.

1.4.3. Caracterizar el objeto donde se quiere aplicar.

1.4.4. Interrelacionar los conocimientos con el objeto.

1.4.5. Elaborar conclusiones.

1.5. evaluar el objeto.

1.6. Aplicar transformaciones en caso de ser necesario.

1. Ejecutar.

1. Explorar.

1.1. Identificar el objeto a explorar.

1.2. Delimitar cada parte del objeto.

1.3. Analizar cada parte del objeto.

1.4. Caracterizar sus partes

2. Manipular.

2.1. Determinar la causa por la cual se utiliza el objeto.

2.2. Determinar las partes del objeto.

2.3. Asociar cada parte del objeto con las acciones a realizar.

2.4. Llevar a cabo la acción.

**Anexo XIII. Relación existente entre sistema de habilidades, contenidos y valores.**

<b>Discip</b>	<b>Asig</b>	<b>Temát.</b>	<b>Habilidades</b>	<b>Habilidades de la disciplina</b>	<b>Habilidades de la carrera</b>		<b>Valores</b>
Operadores de Micro	OPERACIÓN DE COMPUTADORAS-I	Introducción a la Informática	Valorar efectos sociales de las TIC.	Comunicar las ideas y conocimientos de forma eficiente de manera oral o escrita	Caracterizar cualquier aplicación o dispositivo	Conocer	Laboriosidad
			Caracterizar los sistemas de computación.	Comunicar las ideas y conocimientos de forma eficiente de manera oral o escrita	Caracterizar cualquier aplicación o dispositivo	Conocer	Laboriosidad
			Instalar el sistema de computación.	Instalar y configurar el software básico y las aplicaciones.	Instalar y configurar computadoras y sus periféricos.	Preparar	Responsabilidad Laboriosidad

Discip	Asig	Temát.	Habilidades	Habilidades de la disciplina	Habilidades de la carrera		Valores
Operadores de Micro	OPERACIÓN DE COMPUTADORAS-I	Sistemas Operativos	Explotar un sistema operativo ya sea de comandos o de entorno gráfico.	Explotar eficientemente sistemas operativos para computadoras aisladas o conectadas en red.	Ejecutar cambios para mejorar la explotación del sistema y las aplicaciones.	Operar	Responsabilidad Laboriosidad
			Instalar y configurar un sistema operativo.	Instalar y configurar el software básico y las aplicaciones.	Instalar y actualizar aplicaciones informáticas.	Preparar	Responsabilidad Laboriosidad
				Instalar y configurar los servicios de un sistema operativo de servidor dado.			
			Operar con la información en discos.	Explotar las computadoras personales y los sistemas de aplicación.	Ejecutar cambios para mejorar la explotación del sistema y las aplicaciones.	Operar	Responsabilidad Laboriosidad

Discip	Asig	Temát.	Habilidades	Habilidades de la disciplina	Habilidades de la carrera	Valores
--------	------	--------	-------------	------------------------------	---------------------------	---------

peradores de Micro	OPERACIÓN DE COMPUTADORAS-I	Sistemas Operativos	Conectar y los configurar periféricos	Ensamblar de sistema de computadoras aisladas o en red, a partir de sus elementos componentes.	Instalar y configurar computadoras y sus periféricos.	Preparar	Responsa bilidad Laboriosi dad
			Utilizar el ratón y el teclado.	Utilizar con destreza el teclado y el mouse.	Instalar y configurar computadoras y sus periféricos.	Preparar	Responsa bilidad Laboriosi dad
		Seguridad Informática	Caracterizar el plan de seguridad informática.	Comunicar las ideas y conocimientos de forma eficiente de manera oral o escrita	Caracterizar cualquier aplicación o dispositivo	Conocer	Laboriosi dad
			Utilizar programas antivirus.	Instalar y configurar sistemas operativos, sistemas de aplicación, descontaminad ores de virus informáticos y otros software.	Ejecutar el reglamento de seguridad informática.	Mantener	Responsa bilidad Laboriosi dad Compañe rismo

Discip	Asig	Temát.	Habilidades	Habilidades de la disciplina	Habilidades de la carrera	Valores
--------	------	--------	-------------	------------------------------------	---------------------------	---------

Operadores de Micro	OPERACIÓN DE COMPUTADORAS-I	Seguridad Informática	Instalar programas antivirus.	Instalar y configurar el software básico y las aplicaciones.	Instalar y actualizar aplicaciones informáticas.	Preparar	Responsabilidad Laboriosidad
				Instalar y configurar sistemas operativos, sistemas de aplicación, descontaminadores de virus informáticos y otros software.			
			Utilizar e instalar el equipamiento necesario para la protección física y eléctrica.	Instalar y configurar sistemas operativos, sistema de aplicación, descontaminadores de virus informáticos y otros software.	Ejecutar el reglamento de seguridad informática.	Mantener	Responsabilidad Laboriosidad Compañerismo

Discip	Asig	Temát.	Habilidades	Habilidades de la disciplina	Habilidades de la carrera	Valores
--------	------	--------	-------------	------------------------------	---------------------------	---------

Operadores de Micro	OPERACIÓN DE COMPUTADORAS-I	Procesador de texto	Explotar un procesador de texto.	Explotar eficientemente sistemas operativos para computadoras aisladas o conectadas en red.	Ejecutar cambios para mejorar la explotación del sistema y las aplicaciones.	Operar	Responsabilidad Laboriosidad
		Procesador de texto.	Crear, guardar y editar documentos de forma rápida y eficiente.	Explotar eficientemente sistemas operativos para computadoras aisladas o conectadas en red.	Ejecutar cambios para mejorar la explotación del sistema y las aplicaciones.	Operar	Responsabilidad Laboriosidad
			Aplicar el formato adecuado a los documentos.	Explotar las computadoras personales y los sistemas de aplicación	Ejecutar cambios para mejorar la explotación del sistema y las aplicaciones.		Responsabilidad Laboriosidad
			Imprimir documentos.	Explotar las computadoras personales y los sistemas de aplicación	Ejecutar cambios para mejorar la explotación del sistema y las aplicaciones.		Responsabilidad Laboriosidad

Discip	Asig	Temát.	Habilidades	Habilidades de la disciplina	Habilidades de la carrera	Valores
--------	------	--------	-------------	------------------------------	---------------------------	---------

Operadores de Micro	OPERACIÓN DE COMPUTADORAS-I	Hoja de cálculo	Utilizar la ayuda como apoyo en la utilización de esta aplicación.	Explotar las computadoras personales y los sistemas de aplicación	Ejecutar cambios para mejorar la explotación del sistema y las aplicaciones.	Operar	Responsabilidad Laboriosidad
			Utilizar el ratón y el teclado	Utilizar con destreza el teclado y el mouse.	Instalar y configurar computadoras y sus periféricos.	Preparar	Responsabilidad Laboriosidad
			Crear tablas con el uso de fórmulas y funciones	Explotar las computadoras personales y los sistemas de aplicación.	Ejecutar cambios para mejorar la explotación del sistema y las aplicaciones.	Operar	Responsabilidad Laboriosidad
			Vincular fórmulas entre las hojas del libro de trabajo	Explotar las computadoras personales y los sistemas de aplicación.	Ejecutar cambios para mejorar la explotación del sistema y las aplicaciones.	Operar	Responsabilidad Laboriosidad
			Crear tablas con el uso de distintos formatos	Explotar las computadoras personales y los sistemas de aplicación.	Ejecutar cambios para mejorar la explotación del sistema y las aplicaciones.	Operar	Responsabilidad Laboriosidad

Discip	Asig	Temát.	Habilidades	Habilidades de la disciplina	Habilidades de la carrera	Valores
--------	------	--------	-------------	------------------------------	---------------------------	---------



Operadores de Micro	OPERACIÓN DE COMPUTADORAS-I	Hoja de cálculo	Editar o modificar tablas	Explotar las computadoras personales y los sistemas de aplicación.	Ejecutar cambios para mejorar la explotación del sistema y las aplicaciones.	Operar	Responsabilidad Laboriosidad
			Crear y editar gráficos a partir de los datos de la tabla	Explotar las computadoras personales y los sistemas de aplicación.	Ejecutar cambios para mejorar la explotación del sistema y las aplicaciones.	Operar	Responsabilidad Laboriosidad
			Imprimir tablas y gráficos	Explotar las computadoras personales y los sistemas de aplicación	Instalar y configurar computadoras y sus periféricos.	Preparar	Responsabilidad Laboriosidad
			Utilizar el ratón y el teclado	Utilizar con destreza el teclado y el mouse.	Instalar y configurar computadoras y sus periféricos.	Preparar	Responsabilidad Laboriosidad

Discip	Asig	Temát.	Habilidades	Habilidades de la disciplina	Habilidades de la carrera	Valores
--------	------	--------	-------------	------------------------------	---------------------------	---------

Operadores de Micro	OPERACIÓN DE COMPUTADORAS-I	Presentaciones electrónicas	Utilizar la aplicación para comunicar una idea o trabajo a un grupo de personas de forma eficiente	Explotar las computadora s personales y los sistemas de aplicación.	Prestar ayuda especializada para el uso correcto del equipo y las aplicaciones a los usuarios.	Mantener	Responsa bilidad Laboriosi dad Compañe rismo
			Crear presentaciones eficientes y concretas.	Explotar las computadora s personales y los sistemas de aplicación.	Ejecutar cambios para mejorar la explotación del sistema y las aplicaciones.	Operar	Responsa bilidad Laboriosi dad
Operadores de Micro	OPERACIÓN DE COMPUTADORAS-I	Presentaciones electrónicas	Crear diapositivas.	Explotar las computadora s personales y los sistemas de aplicación.	Ejecutar cambios para mejorar la explotación del sistema y las aplicaciones.	Operar	Responsa bilidad Laboriosi dad
			Realizar animaciones en la diapositiva	Explotar las computadora s personales y los sistemas de aplicación.	Ejecutar cambios para mejorar la explotación del sistema y las aplicaciones.	Operar	Responsa bilidad Laboriosi dad

Discip	Asig	Temát.	Habilidades	Habilidades de la disciplina	Habilidades de la carrera	Valores
--------	------	--------	-------------	------------------------------	---------------------------	---------

Operadores de Micro	OPERACIÓN DE COMPUTADORAS-I	Presentaciones electrónicas	Ejecutar presentaciones con animación y sonido	Explotar las computadora s personales y los sistemas de aplicación.	Ejecutar cambios para mejorar la explotación del sistema y las aplicaciones.	Operar	Responsabili dad Laboriosidad
			Utilizar el ratón y el teclado.	Utilizar con destreza el teclado y el mouse.	Instalar y configurar computadoras y sus periféricos.	Preparar	Responsabili dad Laboriosidad
		Conectividad	Explotar las posibilidades que brinda el sistema operativo para manipular información a través de la red	Explotar eficientement e sistemas operativos para computadora s aisladas o conectadas en red.	Utilizar buscadores y navegadores de páginas Web y aplicaciones informáticas.	Operar	Responsabili dad Laboriosidad
			Compartir de forma segura los recursos en la red	Explotar los servicios de Internet y de correo electrónico.	Ejecutar el reglamento de seguridad informática.	Mantener	Responsabili dad Laboriosidad Compañeris mo
			Utilizar el ratón y el teclado.	Utilizar con destreza el teclado y el mouse.	Instalar y configurar computadoras y sus periféricos.	Preparar	Responsabili dad Laboriosidad

Discip	Asig	Temát.	Habilidades	Habilidades de la disciplina	Habilidades de la carrera	Valores
--------	------	--------	-------------	------------------------------	---------------------------	---------

Operadores de Micro	OPERACIÓN DE COMPUTADORAS-I	Internet y correo electrónico	Instalar, actualizar y configurar el cliente de correo electrónico.	Explotar los servicios de Internet y de correo electrónico.	Instalar y actualizar aplicaciones informáticas.	Preparar	Responsabilidad Laboriosidad
				Utilizar buscadores y navegadores. Comparar diferentes arquitecturas de computadoras.	Utilizar buscadores y navegadores de páginas web y aplicaciones informáticas	Operar	Responsabilidad Laboriosidad
				Utilizar los conceptos sobre el funcionamiento de las redes de computadoras para su explotación.	Ejecutar el reglamento de seguridad informática	Mantener	Responsabilidad Laboriosidad Compañerismo Patriotismo
			Utilizar un cliente de correo para enviar y recibir mensajes.	Explotar los servicios de Internet y de correo electrónico.	Utilizar buscadores y navegadores de páginas web y aplicaciones informáticas.	Operar	Responsabilidad Laboriosidad

Discip	Asig	Temát.	Habilidades	Habilidades de la disciplina	Habilidades de la carrera	Valores
--------	------	--------	-------------	------------------------------	---------------------------	---------

Operadores de Micro	OPERACIÓN DE COMPUTADORAS-I	Internet y correo electrónico		Utilizar los conceptos sobre el funcionamiento de las redes de computadoras para su explotación.			
			Instalar, actualizar y configurar el navegador de Internet.		Utilizar buscadores y navegadores de páginas Web y aplicaciones informáticas.	Operar	Responsabilidad Laboriosidad
			Visitar páginas Web.	Utilizar buscadores y navegadores. Comparar diferentes arquitecturas de computadoras.	Utilizar buscadores y navegadores de páginas Web y aplicaciones informáticas.	Operar	Responsabilidad Laboriosidad
			Utilizar buscadores de Internet.	Explotar los servicios de Internet y de correo electrónico.	Utilizar buscadores y navegadores de páginas Web y aplicaciones informáticas.	Operar	Responsabilidad Laboriosidad

Discip	Asig	Temát.	Habilidades	Habilidades de la disciplina	Habilidades de la carrera	Valores
--------	------	--------	-------------	------------------------------	---------------------------	---------

Operadores de Micro	OPERACIÓN DE COMPUTADORAS-I	Internet y correo electrónico	Visitar páginas Web. Utilizar buscadores de Internet.	Utilizar buscadores y navegadores. Comparar diferentes arquitecturas de computadoras	Utilizar buscadores y navegadores de páginas Web y aplicaciones informáticas.	Operar	Responsabilidad Laboriosidad
				Utilizar los conceptos sobre el funcionamiento de las redes de computadoras para su explotación.	Utilizar buscadores y navegadores de páginas Web y aplicaciones informáticas.	Operar	Responsabilidad Laboriosidad
			Utilizar los antivirus sobre los correos recibidos.	Explotar los servicios de Internet y de correo electrónico.	Ejecutar el reglamento de seguridad informática.	Mantener	Responsabilidad Laboriosidad Compañerismo
				Utilizar buscadores y navegadores.			
				Comparar diferentes arquitecturas de computadoras	Ejecutar el reglamento de seguridad informática.	Mantener	Responsabilidad Laboriosidad Compañerismo
			Utilizar el ratón y el teclado.	Utilizar con destreza el teclado y el mouse.	Instalar y configurar computadoras y sus periféricos.	Preparar	Responsabilidad Laboriosidad

Discip	Asig	Temát.	Habilidades	Habilidades de la disciplina	Habilidades de la carrera	Valores
--------	------	--------	-------------	------------------------------	---------------------------	---------

Operadores de Micro	ARQUITECTURA DE COMPUTADORAS I	Introducción a los sistemas digitales	Caracterizar los sistemas digitales.	Comunicar las ideas y conocimientos de forma eficiente de manera oral o escrita	Caracterizar cualquier aplicación o dispositivo	Conocer	Laboriosidad
			Reconocer compuertas lógicas.	Comparar diferentes arquitecturas de computadoras	Caracterizar cualquier aplicación o dispositivo	Conocer	Laboriosidad
			Utilizar la simbología de representación de códigos numéricos.	Comunicar las ideas y conocimientos de forma eficiente de manera oral o escrita	Caracterizar cualquier aplicación o dispositivo	Conocer	Laboriosidad
			Utilizar la simbología de representación de compuertas lógicas.	Comunicar las ideas y conocimientos de forma eficiente de manera oral o escrita	Caracterizar cualquier aplicación o dispositivo	Conocer	Laboriosidad
			Identificar las funciones de circuitos digitales sencillos a base de compuertas lógicas.	Comunicar las ideas y conocimientos de forma eficiente de manera oral o escrita	Caracterizar cualquier aplicación o dispositivo	Conocer	Laboriosidad
		Circuitos lógicos	Identificar las funciones de circuitos combinacionales	Comunicar las ideas y conocimientos de forma eficiente de manera oral o escrita	Caracterizar cualquier aplicación o dispositivo	Conocer	Laboriosidad

Discip	Asig	Temát.	Habilidades	Habilidades de la disciplina	Habilidades de la carrera	Valores
--------	------	--------	-------------	------------------------------	---------------------------	---------

Operadores de Micro	ARQUITECTURA DE COMPUTADORAS I	Circuitos lógicos	Identificar las funciones de los circuitos secuenciales.	Comunicar las ideas y conocimientos de forma eficiente de manera oral o escrita	Caracterizar cualquier aplicación o dispositivo	Conocer	Laboriosidad
		Memorias RAM y ROM	Reconocer los diferentes tipos de memorias.	Comunicar las ideas y conocimientos de forma eficiente de manera oral o escrita	Caracterizar cualquier aplicación o dispositivo	Conocer	Laboriosidad
		Microprocesadores	Seleccionar entre diferentes tipos de microprocesadores	Comparar diferentes arquitecturas de computadoras.	Caracterizar cualquier aplicación o dispositivo	Conocer	Laboriosidad
		Arquitectura de computadoras. Tarjeta principal	Seleccionar entre diferentes tipos de tarjetas principales de computadoras	Comparar diferentes arquitecturas de computadoras.	Caracterizar cualquier aplicación o dispositivo	Conocer	Laboriosidad
			Reconocer las tipos y las fallas de las memorias.	Diagnosticar el origen de las fallas en la computadora y sus periféricos.	Diagnosticar el origen de las fallas en las computadoras y sus periféricos.	Mantener	Responsabilidad Laboriosidad Compañerismo

Discip	Asig	Temát.	Habilidades	Habilidades de la disciplina	Habilidades de la carrera	Valores
--------	------	--------	-------------	------------------------------	---------------------------	---------



Operadores de Micro	ARQUITECTURA DE COMPUTADORAS I	Arquitectura de computadoras. Tarjeta principal	Instalar y configurar un microprocesador en la tarjeta principal de una computadora.	Consultar e interpretar la literatura técnica especializada tanto en español como en inglés.	Ejecutar cambios para mejorar la explotación del sistema y las aplicaciones.	Operar	Responsabilidad Laboriosidad
				Interpretar literatura técnica incluidos manuales en español e inglés sobre la temática de la disciplina.			
			Interpretar la documentación técnica de las tarjetas principales.	Interpretar literatura técnica incluidos manuales en español e inglés sobre la temática de la disciplina.	Ejecutar cambios para mejorar la explotación del sistema y las aplicaciones.	Operar	Responsabilidad Laboriosidad
				Consultar e interpretar la literatura técnica especializada tanto en español como en inglés.			

Discip	Asig	Temát.	Habilidades	Habilidades de la disciplina	Habilidades de la carrera	Valores
--------	------	--------	-------------	------------------------------	---------------------------	---------

Operadores de Micro	ARQUITECTURA DE COMPUTADORAS I	Diagnóstico	Reconocer fallas en la computadora.	Diagnosticar el origen de las fallas en la computadora y sus periféricos.	Diagnosticar el origen de las fallas en las computadoras y sus periféricos.	Mantener	Responsabilidad Laboriosidad Compañerismo
			Diagnosticar el origen de las fallas en la computadora a través de una metodología de diagnóstico.	Diagnosticar el origen de las fallas en la computadora y sus periféricos.	Diagnosticar el origen de las fallas en las computadoras y sus periféricos.	Mantener	Responsabilidad Laboriosidad Compañerismo
Lenguaje y Técnica de Programación	Programación I	Introducción a la teoría semántica del cálculo proposicional	Aplicar las operaciones básicas con conjuntos.	Modelar un fenómeno real desde el punto de vista de la información	Seleccionar herramientas y lenguajes apropiados cuando no se especifique	Implementar	Responsabilidad Laboriosidad
			Representar fórmulas en el cálculo proposicional a partir de situaciones vinculadas a la Ingeniería Informática.	Modelar un fenómeno real desde el punto de vista de la información	Seleccionar herramientas y lenguajes apropiados cuando no se especifique	Implementar	Responsabilidad Laboriosidad
			Construir de Tablas de Verdad.	Modelar un fenómeno real desde el punto de vista de la información	Seleccionar herramientas y lenguajes apropiados cuando no se especifique	Implementar	Responsabilidad Laboriosidad
			Interpretar y clasificar fórmulas.	Modelar un fenómeno real desde el punto de vista de la información	Seleccionar herramientas y lenguajes apropiados cuando no se especifique	Implementar	Responsabilidad Laboriosidad

Discip	Asig	Temát.	Habilidades	Habilidades de la disciplina	Habilidades de la carrera	Valores
--------	------	--------	-------------	------------------------------	---------------------------	---------

Lenguaje y Técnica de Programación	Programación I	Algoritmización	Modelar problemas que se resuelvan con algoritmos de secuencia lineal o con alternativas en una de las representaciones estudiadas.	Modelar un fenómeno real desde el punto de vista de la información	Codificar algoritmos y aplicaciones (generales, Web y Bases de Datos)	Implementar	Responsabilidad Laboriosidad
			Utilizar diferentes tipos de variables en la modelación de problemas y establecer Precondiciones y Poscondiciones para las mismas.	Explotar los servicios de Internet y de correo electrónico.	Proponer, coordinar y ejecutar cambios para mejorar la funcionalidad del software.	Mantener	Responsabilidad Laboriosidad Compañerismo
			Desarrollar algoritmos con iteraciones simples y anidadas; en combinación con otras estructuras estudiadas en la resolución de aquellos problemas que lo requieran.	Codificar algoritmos para aplicaciones generales y de bases de datos	Propone y coordina cambios para mejorar la explotación del sistema y las aplicaciones	Controlar	Responsabilidad Laboriosidad
					Codificar algoritmos y aplicaciones (generales, Web y Bases de Datos)	Implementar	Responsabilidad Laboriosidad

Discip	Asig	Temát.	Habilidades	Habilidades de la disciplina	Habilidades de la carrera	Valores
--------	------	--------	-------------	------------------------------	---------------------------	---------

Lenguaje y Técnica de Programación	Programación I	Algoritmización	Probar los algoritmos utilizando conjuntos de datos apropiados (rastrear).	Comentar y documentar programas y sistemas	Utilizar juegos de datos necesarios y herramientas para probar el software.	Depurar	Responsabilidad Laboriosidad
			Evaluar la eficiencia de los algoritmos.	Propone, coordina y ejecuta cambios para mejorar la funcionalidad del software.	Propone y ejecuta cambios para mejorar la funcionalidad del software	Depurar	Responsabilidad Laboriosidad
		Introducción a la programación	Utilizar adecuadamente las sentencias de asignación, entrada y salida, alternativas simples, además de realizar adecuadamente la declaración de variables y las operaciones aritméticas entre ellas.	Codificar algoritmos para aplicaciones generales y de bases de datos	Comentar y documentar el código	Implementar	Responsabilidad Laboriosidad
			Implementar y poner a punto programas para solucionar problemas de secuencia lineal y con alternativas simples en un lenguaje de alto nivel haciendo uso de buenas prácticas de programación.	Codificar algoritmos para aplicaciones generales y de bases de datos	Poner a punto el software.	Depurar	Responsabilidad Laboriosidad

Discip	Asig	Temát.	Habilidades	Habilidades de la disciplina	Habilidades de la carrera	Valores
--------	------	--------	-------------	------------------------------	---------------------------	---------

Lenguaje y Técnica de Programación	Programación I	Introducción a la programación	Interpretar los algoritmos de secuencia lineal y con alternativas simples programados por otros, así como evaluarlos siguiendo manualmente la traza de la ejecución de los algoritmos.	Valorar críticamente el proceso seguido y los resultados obtenidos.	Utilizar juegos de datos necesarios y herramientas para probar el software.	Depurar	Responsabilidad Laboriosidad
					Diagnosticar el origen de las fallas en el software.		
		Estilos de códigos	Conocer los principales conceptos relacionados con los estilos de códigos en la programación	Obtener una visión general de las distintas fases del desarrollo del software	Valorar los elementos a tener en cuenta para el desarrollo de un proyecto.	Conocer	Responsabilidad Laboriosidad
			Conocer los beneficios que aportan los estilos de codificación a las comunidades de programadores y su influencia en la calidad del producto final.	Obtener una visión general de las distintas fases del desarrollo del software	Valorar los elementos a tener en cuenta para el desarrollo de un proyecto.	Conocer	Responsabilidad Laboriosidad
			Codificar un ejercicio de programación utilizando un estilo de código determinado	Codificar algoritmos para aplicaciones generales y de bases de datos	Poner a punto el software.	Depurar	Responsabilidad Laboriosidad
			Interpretar un ejercicio de programación codificado con un estilo determinado	Valorar críticamente el proceso seguido y los resultados obtenidos.	Valorar los elementos a tener en cuenta para el desarrollo de un proyecto.	Conocer	Responsabilidad Laboriosidad

Discip	Asig	Temát.	Habilidades	Habilidades de la disciplina	Habilidades de la carrera	Valores
--------	------	--------	-------------	------------------------------	---------------------------	---------

Lenguaje y Técnica de Programación	Programación I		Conocer algunos principios para el diseño de un estilo de código	Obtener una visión general de las distintas fases del desarrollo del software	Valorar los elementos a tener en cuenta para el desarrollo de un proyecto.	Conocer	Responsabilidad Laboriosidad
			Diseñar un estilo de codificación	Obtener la experiencia de trabajar en un proyecto real.	Valorar los elementos a tener en cuenta para el desarrollo de un proyecto.	Conocer	Responsabilidad Laboriosidad
			Codificar un ejercicio con un estilo de código diseñado por los estudiantes	Codificar algoritmos para aplicaciones generales y de bases de datos	Valorar los elementos a tener en cuenta para el desarrollo de un proyecto.	Conocer	Responsabilidad Laboriosidad
			Compara el estilo de codificación diseñado con un conjunto de principios a seguir.	Valorar críticamente el proceso seguido y los resultados obtenidos.	Valorar los elementos a tener en cuenta para el desarrollo de un proyecto.	Conocer	Responsabilidad Laboriosidad
		Pruebas.	Definir que son las pruebas de un software	Obtener la experiencia de trabajar en un proyecto real.	Valorar los elementos a tener en cuenta para el desarrollo de un proyecto.	Conocer	Responsabilidad Laboriosidad
			Explicar en que consisten las pruebas de caja blanca y caja negra.	Obtener la experiencia de trabajar en un proyecto real.	Valorar los elementos a tener en cuenta para el desarrollo de un proyecto.	Conocer	Responsabilidad Laboriosidad
			Elaborar casos de prueba sencillos.	Obtener la experiencia de trabajar en un proyecto real.	Valorar los elementos a tener en cuenta para el desarrollo de un proyecto.	Conocer	Responsabilidad Laboriosidad
			Definir qué es la depuración	Obtener la experiencia de trabajar en un proyecto real.	Valorar los elementos a tener en cuenta para el desarrollo de un proyecto.	Conocer	Responsabilidad Laboriosidad

Discip	Asig	Temát.	Habilidades	Habilidades de la disciplina	Habilidades de la carrera	Valores
--------	------	--------	-------------	------------------------------	---------------------------	---------

Lenguaje y Técnica de Programación	Programación I	Pruebas.	Diseñar casos de prueba empleando la técnica de partición de equivalencia	Obtener la experiencia de trabajar en un proyecto real.	Valorar los elementos a tener en cuenta para el desarrollo de un proyecto.	Conocer	Responsabilidad Laboriosidad
			Ejecutar pruebas de software	Valorar críticamente el proceso seguido y los resultados obtenidos.	Valorar los elementos a tener en cuenta para el desarrollo de un proyecto.	Conocer	Responsabilidad Laboriosidad
			Definir que es la depuración de errores	Valorar críticamente el proceso seguido y los resultados obtenidos.	Valorar los elementos a tener en cuenta para el desarrollo de un proyecto.	Conocer	Responsabilidad Laboriosidad
Proyectos Informáticos	Proyectos Informáticos I	Proyectos Informáticos	Saber identificar las características que definen la naturaleza de un proyecto.	Obtener una visión general de las distintas fases del desarrollo del software	Valorar los elementos a tener en cuenta para el desarrollo de un proyecto.	Conocer	Responsabilidad Laboriosidad
			Identificar cuáles son los componentes básicos sobre los que debe descansar cualquier tipo de proyecto.	Obtener una visión general de las distintas fases del desarrollo del software	Valorar los elementos a tener en cuenta para el desarrollo de un proyecto.	Conocer	Responsabilidad Laboriosidad
			Ilustrar las diferentes fases que componen el proceso de gestión de un proyecto.	Obtener una visión general de las distintas fases del desarrollo del software	Valorar los elementos a tener en cuenta para el desarrollo de un proyecto.	Conocer	Responsabilidad Laboriosidad
			Evaluar la viabilidad de un proyecto.	Valorar críticamente el proceso seguido y los resultados obtenidos.	Valorar los elementos a tener en cuenta para el desarrollo de un proyecto.	Conocer	Responsabilidad Laboriosidad

Discip	Asig	Temát.	Habilidades	Habilidades de la disciplina	Habilidades de la carrera	Valores
--------	------	--------	-------------	------------------------------	---------------------------	---------

Proyectos Informáticos	Proyectos Informáticos I	Proyectos Informáticos	Valorar las características que deben tener los objetivos de un proyecto para que sean efectivos.	Valorar críticamente el proceso seguido y los resultados obtenidos.	Valorar los elementos a tener en cuenta para el desarrollo de un proyecto.	Conocer	Responsabilidad Laboriosidad
			Analizar aquellos elementos que han de tenerse en cuenta para preparar adecuadamente la planificación de un proyecto.	Valorar críticamente el proceso seguido y los resultados obtenidos.	Valorar los elementos a tener en cuenta para el desarrollo de un proyecto.	Conocer	Responsabilidad Laboriosidad
			Conocer algunas herramientas que facilitan el seguimiento y control de su proyecto.	Obtener una visión general de las distintas fases del desarrollo del software	Valorar los elementos a tener en cuenta para el desarrollo de un proyecto.	Conocer	Responsabilidad Laboriosidad
		Introducción al PSP	Identificar los pasos del proceso de mejora.	Obtener una visión general de las distintas fases del desarrollo del software	Valorar los elementos a tener en cuenta para el desarrollo de un proyecto.	Conocer	Responsabilidad Laboriosidad
			Valorar la necesidad de gestionar y controlar cómo utilizar el tiempo para su mejor aprovechamiento.	Valorar críticamente el proceso seguido y los resultados obtenidos.	Valorar los elementos a tener en cuenta para el desarrollo de un proyecto.	Conocer	Responsabilidad Laboriosidad
			Valorar cómo utilizar los datos registrados para hacer planificaciones de períodos y productos.	Obtener una visión general de las distintas fases del desarrollo del software	Valorar los elementos a tener en cuenta para el desarrollo de un proyecto.	Conocer	Responsabilidad Laboriosidad

Discip	Asig	Temát.	Habilidades	Habilidades de la disciplina	Habilidades de la carrera	Valores
--------	------	--------	-------------	------------------------------	---------------------------	---------



Proyectos Informáticos	Proyectos Informáticos I	Introducción al PSP	Estimar el tamaño de los trabajos que deben realizar.	Obtener una visión general de las distintas fases del desarrollo del software	Valorar los elementos a tener en cuenta para el desarrollo de un proyecto.	Conocer	Responsabilidad Laboriosidad
			Gestionar el tiempo, siendo capaces de decidir como reajustar lo planificado para cumplir con lo estimado.	Obtener una visión general de las distintas fases del desarrollo del software	Valorar los elementos a tener en cuenta para el desarrollo de un proyecto.	Conocer	Responsabilidad Laboriosidad
			Gestionar los compromisos para evitar romperlos u olvidarlos	Obtener una visión general de las distintas fases del desarrollo del software	Valorar los elementos a tener en cuenta para el desarrollo de un proyecto.	Conocer	Responsabilidad Laboriosidad
			Utilizar las programaciones para controlar el progreso de un proyecto.	Valorar críticamente el proceso seguido y los resultados obtenidos.	Valorar los elementos a tener en cuenta para el desarrollo de un proyecto.	Conocer	Responsabilidad Laboriosidad
			Realizar planes de proyectos.	Obtener una visión general de las distintas fases del desarrollo del software	Valorar los elementos a tener en cuenta para el desarrollo de un proyecto.	Conocer	Responsabilidad Laboriosidad

Discip	Asig	Temát.	Habilidades	Habilidades de la disciplina	Habilidades de la carrera	Valores
--------	------	--------	-------------	------------------------------	---------------------------	---------

Proyectos Informáticos	Proyectos Informáticos I	La comunicación en el trabajo colaborativo.	Valorar la importancia de la comunicación en las relaciones interpersonales para el desarrollo de proyectos que requieren del trabajo colaborativo, donde sus integrantes deben mantener una ética profesional para el exitoso alcance de su misión.	Cooperar en equipos de trabajo.	Valorar los elementos a tener en cuenta para el desarrollo de un proyecto.	Conocer	Responsabilidad	Laboriosidad		
			El guión.	Identificar los pasos constitutivos de la estructura de un guión para la elaboración de productos informáticos.	Obtener una visión general de las distintas fases del desarrollo del software		Valorar los elementos a tener en cuenta para el desarrollo de un proyecto.	Conocer	Responsabilidad	Laboriosidad
		Elementos de metodología de la investigación.	Valorar la importancia de un buen fichado a la hora de recopilar información.	Obtener una visión general de las distintas fases del desarrollo del software	Valorar los elementos a tener en cuenta para el desarrollo de un proyecto.	Conocer	Responsabilidad	Laboriosidad		
			Determinar la metodología a seguir para aplicar los diferentes instrumentos para la búsqueda de la información.	Obtener una visión general de las distintas fases del desarrollo del software	Valorar los elementos a tener en cuenta para el desarrollo de un proyecto.	Conocer	Responsabilidad	Laboriosidad		

Discip	Asig	Temát.	Habilidades	Habilidades de la disciplina	Habilidades de la carrera	Valores
--------	------	--------	-------------	------------------------------	---------------------------	---------

Proyectos Informáticos	Proyectos Informáticos I	Eleven. de metodol.inv.	Caracterizar las partes del informe para la presentación de un proyecto.	Obtener una visión general de las distintas fases del desarrollo del software	Valorar los elementos a tener en cuenta para el desarrollo de un proyecto.	Conocer	Responsabilidad Laboriosidad
Tratamiento gráfico e informacional	Tratamiento gráfico e informacional I	Infografía.	Identificar los tipos de colores y sus significados, combinación de colores, contrastes.	Obtener una visión general de las distintas fases del desarrollo del software	Valorar los elementos a tener en cuenta para el desarrollo de un proyecto.	Conocer	Responsabilidad Laboriosidad
			Determinar cómo relacionar colores con contenidos.	Obtener una visión general de las distintas fases del desarrollo del software	Valorar los elementos a tener en cuenta para el desarrollo de un proyecto.	Conocer	Responsabilidad Laboriosidad
		Tratamiento de textos.	Tipos de textos, formatos.	Explotar las computadoras personales y los sistemas de aplicación.	Ejecutar cambios para mejorar la explotación del sistema y las aplicaciones.	Operar	Responsabilidad Laboriosidad
			Aplicar los diferentes formatos de textos en la aplicación Microsoft Word	Explotar las computadoras personales y los sistemas de aplicación.	Ejecutar cambios para mejorar la explotación del sistema y las aplicaciones.	Operar	Responsabilidad Laboriosidad
			Determinar las herramientas necesarias para el tratamiento de textos en la aplicación Photoshop	Explotar las computadoras personales y los sistemas de aplicación.	Ejecutar cambios para mejorar la explotación del sistema y las aplicaciones.	Operar	Responsabilidad Laboriosidad

Discip	Asig	Temát.	Habilidades	Habilidades de la disciplina	Habilidades de la carrera	Valores
--------	------	--------	-------------	------------------------------	---------------------------	---------

Tratamiento gráfico e informacional	Tratamiento gráfico e informacional I	Tratamiento de textos.	Aplicar los diferentes filtros que ya se conocen a las transformaciones de textos.	Explotar las computadoras personales y los sistemas de aplicación.	Ejecutar cambios para mejorar la explotación del sistema y las aplicaciones.	Operar	Responsabilidad Laboriosidad
		Tratamiento de imágenes.	Determinar los diferentes tipos de imágenes y las diferentes transformaciones que se pueden aplicar a las mismas	Explotar las computadoras personales y los sistemas de aplicación.	Ejecutar cambios para mejorar la explotación del sistema y las aplicaciones.	Operar	Responsabilidad Laboriosidad
			Determinar las herramientas necesarias para el tratamiento de imágenes en la aplicación Photoshop	Explotar las computadoras personales y los sistemas de aplicación.	Ejecutar cambios para mejorar la explotación del sistema y las aplicaciones.	Operar	Responsabilidad Laboriosidad
			Aplicar diferentes transformaciones a imágenes a través de Photoshop.	Explotar las computadoras personales y los sistemas de aplicación.	Ejecutar cambios para mejorar la explotación del sistema y las aplicaciones.	Operar	Responsabilidad Laboriosidad
			Aplicar los diferentes filtros que ya se conocen a las transformaciones de imágenes.	Explotar las computadoras personales y los sistemas de aplicación.	Ejecutar cambios para mejorar la explotación del sistema y las aplicaciones.	Operar	Responsabilidad Laboriosidad
		Dream Weaver.	Introducción al Dream Weaver, componentes y propiedades.	Explotar las computadoras personales y los sistemas de aplicación.	Ejecutar cambios para mejorar la explotación del sistema y las aplicaciones.	Operar	Responsabilidad Laboriosidad

Discip	Asig	Temát.	Habilidades	Habilidades de la disciplina	Habilidades de la carrera	Valores
--------	------	--------	-------------	------------------------------	---------------------------	---------

Tratamiento gráfico e informacional	Tratamiento gráfico e informacional I	Dream Weaver.	Utilizar herramientas a partir de los conceptos básicos del tema Web	Explotar las computadoras personales y los sistemas de aplicación.	Ejecutar cambios para mejorar la explotación del sistema y las aplicaciones.	Operar	Responsabilidad Laboriosidad
			Elaborar el guión para confeccionar un sitio web.	Obtener una visión general de las distintas fases del desarrollo del software	Valorar los elementos a tener en cuenta para el desarrollo de un proyecto.	Conocer	Responsabilidad Laboriosidad
			Elaborar sitios que propicien el trabajo independiente y en grupo de los estudiantes con los diseños organizados y estéticamente aceptables.	Explotar las computadoras personales y los sistemas de aplicación.	Ejecutar cambios para mejorar la explotación del sistema y las aplicaciones.	Operar	Responsabilidad Laboriosidad
			Crear vínculos a través de los objetos interactivos.	Explotar las computadoras personales y los sistemas de aplicación.	Ejecutar cambios para mejorar la explotación del sistema y las aplicaciones.	Operar	Responsabilidad Laboriosidad
			Caracterizar la utilidad de las tablas y las plantillas a través de la creación de estas.	Explotar las computadoras personales y los sistemas de aplicación.	Ejecutar cambios para mejorar la explotación del sistema y las aplicaciones.	Operar	
			Confeccionar y evaluar un sitio Web	Explotar las computadoras personales y los sistemas de aplicación.	Ejecutar cambios para mejorar la explotación del sistema y las aplicaciones.	Operar	Responsabilidad Laboriosidad

Discip	Asig	Temát.	Habilidades	Habilidades de la disciplina	Habilidades de la carrera	Valores
--------	------	--------	-------------	------------------------------	---------------------------	---------

Tratamiento gráfico e informacional	Tratamiento gráfico e informacional I	Robohelp.	Determinar la estructura que debe tener una ayuda, su utilidad.	Obtener una visión general de las distintas fases del desarrollo del software	Valorar los elementos a tener en cuenta para el desarrollo de un proyecto.	Conocer	Responsabilidad Laboriosidad
			Introducción al robohelp, componentes.	Explotar las computadoras personales y los sistemas de aplicación.	Ejecutar cambios para mejorar la explotación del sistema y las aplicaciones.	Operar	Responsabilidad Laboriosidad
			Utilizar las diferentes herramientas a partir de los conceptos básicos del tema	Explotar las computadoras personales y los sistemas de aplicación.	Ejecutar cambios para mejorar la explotación del sistema y las aplicaciones.	Operar	Responsabilidad Laboriosidad
			Confeccionar y evaluar una ayuda para una Web.	Explotar las computadoras personales y los sistemas de aplicación.	Ejecutar cambios para mejorar la explotación del sistema y las aplicaciones.	Operar	Responsabilidad Laboriosidad
			Vincular la ayuda a la web asociada a él.	Explotar las computadoras personales y los sistemas de aplicación.	Ejecutar cambios para mejorar la explotación del sistema y las aplicaciones.	Operar	Responsabilidad Laboriosidad

## **Anexo XVI. Actividad metodológica con los jefes de departamentos de las asignaturas técnicas del Politécnico de Informática Carlos Hidalgo Díaz.**

### **Objetivo:**

Determinar la relación existente entre habilidades – disciplinas y el tipo de ejercicio a elaborar en función de esta relación, partiendo del análisis de nuestro objetivo principal, contribuyendo a desarrollar el pensamiento lógico.

### **Introducción.**

La manera en que el proceso de aprendizaje se comprende determina sin dudas cómo se concibe el sentido de la enseñanza y de la intervención educativa, y refleja, a su vez, una concepción del ser humano, de su personalidad y del proceso de su desarrollo, que son las claves (explícitas o implícitas) de cualquier proyecto educativo.

Durante el desarrollo del curso escolar (2004-2005), los profesores que imparten la asignatura Lenguaje y Técnica de Programación se enfrentaron a serias dificultades en cuanto a la asimilación de los contenidos por parte de los alumnos, principalmente los de primer año.

En la búsqueda de los factores que la propiciaban se realizó una entrevista a profesores con el objetivo de determinar los factores que hacen posible el aprendizaje insuficiente de los alumnos, en la cual evaluaríamos las categorías: conocimiento del perfil del egresado, dominio de las habilidades, preparación de los profesores y aprendizaje de los alumnos.

¿Pero, sería este el mismo criterio de los alumnos?

Para dar respuesta a esta interrogante, se realizó una entrevista a los alumnos con el objetivo de valorar la motivación que tienen los estudiantes para el estudio de la Informática y el conocimiento que poseen sobre lo que deben saber hacer como futuros Técnicos medios en Informática, en la cual evaluaríamos las categorías: motivación de los alumnos, conocimiento del perfil del egresado y aprendizaje de los alumnos

¿Cuán real serían estas causas?

Para corroborar la situación presentada se aplicaron los siguientes instrumentos:

1. Encuesta a profesores
2. Encuesta a alumnos.
3. Observación a clases.

#### 4. Análisis de documentos.

Al analizar el resultado de los instrumentos aplicados, derivamos como principal

##### **Problema:**

Insuficiente preparación de los profesores, en el tratamiento de las habilidades informáticas que contribuyen al desarrollo del pensamiento lógico de los alumnos, en el proceso docente educativo de la Computación en el Politécnico de Informática de Pinar del Río.

¿Pero qué es el pensamiento lógico?

Después de haber realizado un análisis de las bases filosóficas, psicológicas y pedagógicas de este concepto se llega a la conclusión de que

De forma general entendemos por pensamiento lógico al proceso que pone en práctica una secuencia de acciones organizadas según su complejidad y que tienen implícitos en ellas la formulación de ideas, juicios y conceptos, así como la relación entre los mismos. (La autora)

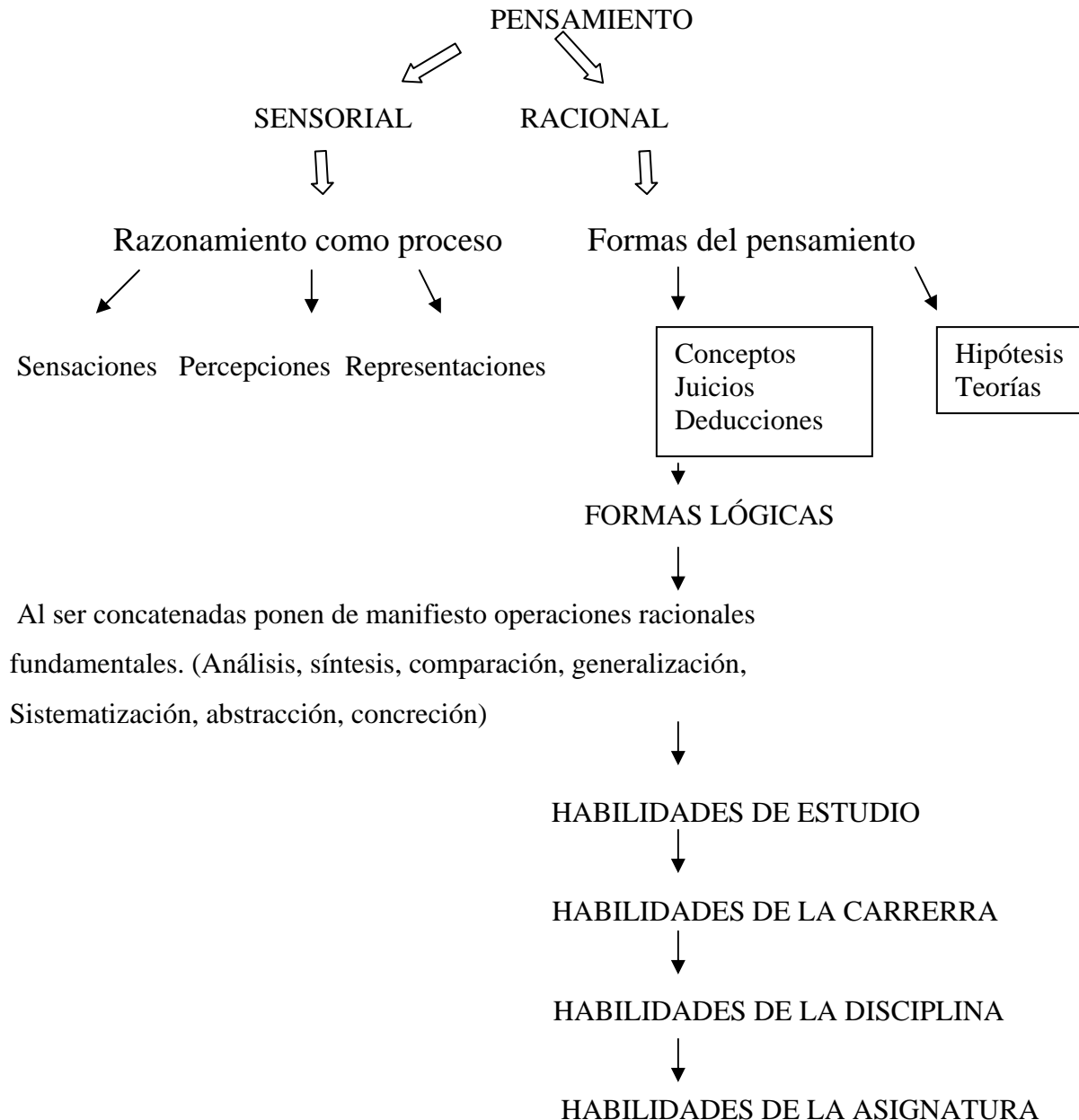
¿El poco desarrollo del pensamiento lógico afecta solo a esta asignatura?

¿Hasta qué punto, el resto de las asignaturas técnicas, pueden influir en la formación de estos jóvenes programadores, comprometidos con la patria? (OHO)



## Desarrollo.

Veamos la siguiente relación:



¿Qué es una habilidad y más específico qué es una habilidad informática?

Para llegar a definir este concepto se hizo un análisis similar al anterior, de lo que se dedujo:

Al analizar todos los aspectos abordados en este epígrafe podemos decir que una habilidad informática son actividades conscientes que realiza el individuo en las que intervienen un sistema de acciones, operaciones y conocimientos, que propician el desarrollo de capacidades y que van encaminadas al uso y explotación adecuado del Hardware y el Software. (La autora)

O sea que para formar el pensamiento lógico de los alumnos hay que desarrollar habilidades y las habilidades a formar las encontramos en los programas de estudio, de ahí que tengan que ser analizados.

Si tenemos en cuenta los términos de diseño curricular y las fases de elaboración nos encontramos con que:

Por qué y para qué enseñar

Qué enseñar – aprender

Cuándo y cómo enseñar – aprender

Qué, cuándo y cómo evaluar.

Existen elementos en el currículo del cual se habla, que no satisfacen condiciones necesarias que deben aparecer como por ejemplo:

1. Los programas de las diferentes disciplinas solo contienen los aspectos a tratar en el primer semestre de la carrera lo que impide que el profesor pueda trazarse metas futuras o establecer nexos de los contenidos que trata con los que están por manipularse y menos aún orientar trabajos investigativos que propicien el aprendizaje por proyectos, aspecto este que es la raíz del tipo de modelo de estudiante que se quiere formar en nuestra enseñanza.
2. No se tienen en cuenta las motivaciones de los estudiantes para definir los objetivos a cumplir en los programas.
3. Las disciplinas de Diseño y Programación, a pesar de tributar ambas a la confección de los Software a producir por estudiantes y profesores, se tratan de forma divorciada.

4. A pesar de que los estudiantes comienzan con sus proyectos en primer año, ellos reciben la asignatura diseño a partir del segundo y es la asignatura de Proyectos Informáticos la que debe asumir esta responsabilidad de forma muy ligera.
5. Se implementan en un mismo curso escolar más de un lenguaje de programación, lo que posibilita que el estudiante no profundice en ninguno, o sea, sabrá un poco de todo, pero no dominará un lenguaje determinado.
6. Existe un número inmenso de habilidades a desarrollar en el programa de cada disciplina, aspecto este que se hace imposible durante un curso escolar y es por esto que a veces se dice tenemos los objetivos tratados pero no vencidos.

Para el resto de las dificultades hay un tratamiento en específico, pero el más importante a tratar es el último, pues como vimos en la relación anterior es el que más nos afecta.

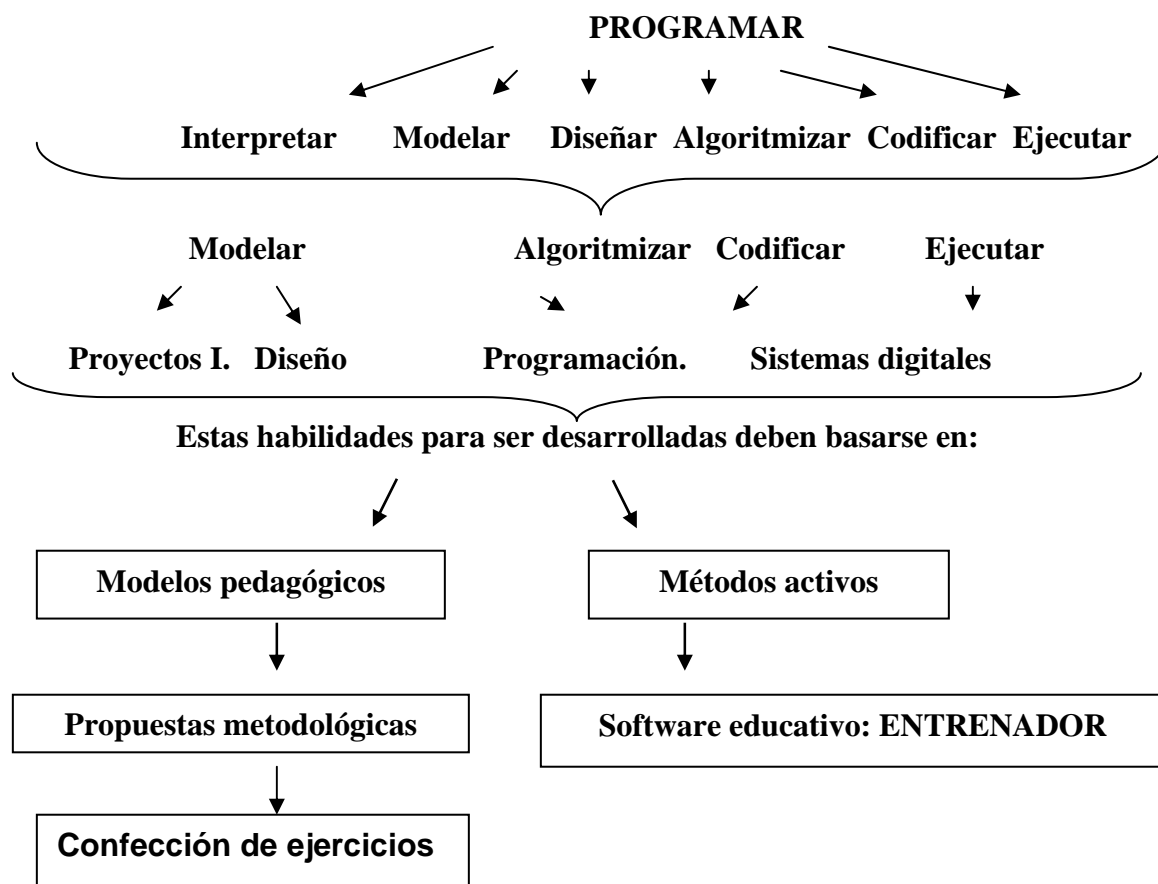
Comenzar a formar esquema que relaciona habilidad-disciplina.

¿Cuál es la misión del centro?

¿Cuándo podemos decir que nuestros estudiantes saben programar?

¿Qué disciplina se relaciona con cada habilidad interna?

Desglosemos ahora cada habilidad de cada asignatura.



¿Qué características deben tener nuestros ejercicios?

1. Ordenados según su nivel de complejidad.
2. Elaborados según la habilidad a desarrollar en la disciplina.
3. Encaminados a la elaboración de respuestas lógicas, propiciando el desarrollo de este tipo de pensamiento.
4. Complejos en su estructura por las operaciones a realizar en ellos.
5. En ellos se exige una nueva forma de respuesta.
6. Fáciles de abordar de forma teórica, lo que contribuye a su tratamiento sin interrupciones por problemas de tecnologías.
7. Previstos para ser tratados según los niveles de aprendizaje.
8. Relacionados con la vida práctica.

Veamos algunos ejemplos:

1. Elaborados según la habilidad a desarrollar en la disciplina.
2. Complejos en su estructura por las operaciones a realizar en ellos.

### **Interpretar – Proyectos Informáticos.**

1. Al departamento de Proyectos Informáticos de nuestro centro se le asignó dar respuesta a una problemática planteada por la Empresa Confecciones ALBA de Pinar del Río. La misma no contaba con un programa que divulgara la información relacionada con el trabajo que realiza la institución.

- a) Seleccione de forma ordenada, cuáles son los pasos a seguir para dar solución al problema.

- ☐ Definir el objetivo
- ☐ Programar
- ☐ Definir los medios necesarios.
- ☐ Definir las necesidades del cliente
- ☐ Definir las necesidades del usuario.
- ☐ Definir tiempo de producción.
- ☐ Determinar la aplicación a utilizar.
- ☐ Confeccionar el guión.
- ☐ Diseñar.

- b) Complete las siguientes frases según sus criterios acerca de la situación planteada.

- Como objetivo se propone: \_\_\_\_\_ la información relacionada con la empresa Confecciones ALBA a través de un \_\_\_\_\_.
- En el diseño del proyecto deben aparecer \_\_\_\_\_ de la institución, así como de cada uno de sus \_\_\_\_\_.
- El proyecto debe reunir todos los requisitos que desee el \_\_\_\_\_.

3. Encaminados a la elaboración de respuestas lógicas, propiciando el desarrollo de este tipo de pensamiento.
4. Relacionados con la vida práctica.

### **Modelar- Proyectos Informáticos y Diseño.**

1. Se quiere confeccionar el diseño de una casa de campaña que se producirá para la venta a los campismos del país. Seleccione cronológicamente los pasos a seguir para dar respuesta a esta propuesta.

- \_\_\_\_\_ Seleccionar las diferentes formas de la casa de campaña.
- \_\_\_\_\_ Seleccionar los colores que tendrá las casas.
- \_\_\_\_\_ Investigar la características de las zonas geográficas en los que se encuentran ubicados los campismos.
- \_\_\_\_\_ Investigar las diferentes características de los centros que harán las compras.
- \_\_\_\_\_ Determinar la capacidad que tendrán las casas.
- \_\_\_\_\_ Buscar nombre del director del campismo.
- \_\_\_\_\_ Definir la utilización de un logotipos y/o imágenes en la tela y su ubicación.
- \_\_\_\_\_ Investigar los colores de los catres que repartirán.
- \_\_\_\_\_ Determinar cantidad de actividades recreativas.

5. En ellos se exige una nueva forma de respuesta.

Aquí se pone de manifiesto el tipo de ejercicio a los que se va a enfrentar nuestros estudiantes.

- De desarrollo.
- De construcción.
- De completamiento.
- De selección.

De construcción.

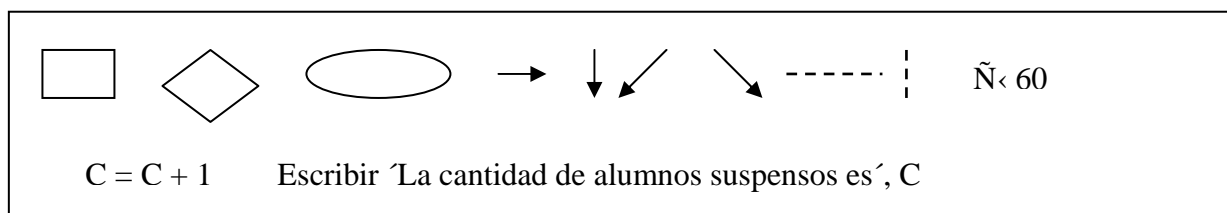
**Algoritmizar – Programación.**

2. La secretaria docente de la escuela ha pedido al jefe de departamento de la asignatura Proyectos Informáticos, la cantidad de alumnos suspensos en el primer trabajo práctico.

a) Complete el siguiente algoritmo que se propone para dar respuesta a esta petición si se conoce que el grupo tiene 28 alumnos.

- Inicio
- 
- Leer 'Nota', N
- 
- 
- Fin

b) Represente el algoritmo planteado a través de un diagrama de bloques, seleccionando las imágenes de la propuesta dada.



6. Fáciles de abordar de forma teórica, lo que contribuye a su tratamiento sin interrupciones por problemas de tecnologías.

7. Previstos para ser tratados según los niveles de aprendizaje.

- Nivel reproductivo.
- Nivel aplicativo.

- Nivel creativo.

#### **Nivel reproductivo.**



Se le daría el ejercicio anterior con más elementos de ayuda o sin el segundo inciso

#### **Nivel aplicativo.**



Se le daría el ejercicio anterior con menos elementos de ayuda o así como está.

#### **Nivel creativo**



Se le daría el ejercicio anterior sin elementos de ayuda en el los inciso.

¿Qué cantidad de ejercicios debemos elaborar?

Esto depende del tipo de contenido a tratar y el tipo de estudiante con los que trabajamos, de ahí que sea el profesor el que lo defina.

#### **Conclusiones.**

El aprendizaje de nuestros estudiantes depende en gran medida de la seriedad con que nos preparemos para desarrollarlo y en él está la base donde radica cada uno de los problemas con que nos enfrentamos cada día como educadores.

Les propongo a que la próxima actividad metodológica a desarrollar en sus departamentos, después de esta, sea el debate del sistema de ejercicios creados por ustedes mismos, recordando siempre las características que deben tener, las habilidades que deben desarrollar y los contenidos que tratarán en ellos.



## **Anexo XV. Clase metodológica para el correcto uso del software**

Para la puesta en marcha de esta clase metodológica se toma como regularidad: El poco desarrollo del pensamiento lógico de los alumnos del IPI:”Carlos Hidalgo Díaz”

**Objetivo:** Capacitar a los profesores en el uso y explotación de PENSOLOG, partiendo del análisis de los componentes de una clase, contribuyendo a desarrollar el pensamiento lógico de los alumnos.

**Asignatura:** Proyectos Informáticos I.

**Unidad 1.** Introducción a la disciplina Proyectos Informáticos.

La unidad a la cual hacemos referencia tiene un total de 12 h/c divididas en dos clases de 3 h/c cada una.

En cada una de estas clases se orientan ejercicios a resolver de forma independiente para lo cual deberán auxiliarse como material bibliográfico de nuestro sistema (medio de enseñanza), así mismo sucederá con la resolución de tareas extractases.

Los alumnos y profesores pueden acceder al sitio diseñado en su tiempo libre para su preparación personal o como un ejemplo a analizar para introducir el estudio de otros contenidos.

### **Derivación gradual de los objetivos.**

#### **Objetivos de la carrera.**

- ♣ Valorar los elementos a tener en cuenta para el desarrollo de un proyecto.

#### **Objetivos de la disciplina.**

- ♣ Obtener una visión general de las distintas fases del desarrollo del software
- ♣ Valorar críticamente el proceso seguido y los resultados obtenidos.

#### **Objetivo de la asignatura.**

- ♣ Identificar las características que definen la naturaleza de un proyecto.
- ♣ Identificar cuáles son los componentes básicos sobre los que debe descansar cualquier tipo de proyecto.
- ♣ Ilustrar las diferentes fases que componen el proceso de gestión de un proyecto.

- ♣ Evaluar la viabilidad de un proyecto.
- ♣ Valorar las características que deben tener los objetivos de un proyecto para que sean efectivos.
- ♣ Analizar aquellos elementos que han de tenerse en cuenta para preparar adecuadamente la planificación de un proyecto.
- ♣ Conocer algunas herramientas que facilitan el seguimiento y control de su proyecto.

## **Objetivo de las clases.**

### **Contenidos.**

#### **Clase 1.**

- ♣ Definición de Informática.
- ♣ Antecedentes históricos de la especialidad.
- ♣ Plan de estudio. Disciplinas y asignaturas.
- ♣ Perfil del egresado.
- ♣ Objetivos de la carrera o la especialidad.
- ♣ Habilidades técnicas generales y específicas.

#### **Clase 2.**

- ♣ Introducción a la disciplina Proyectos Informáticos y a la asignatura Proyectos Informáticos I.
- ♣ Definición de proyecto.
- ♣ Definición de proyecto informático.
- ♣ Características de un proyecto.
- ♣ Ciclo de vida de un proyecto.

#### **Clase 3.**

- ♣ Fases de un proyecto.
- ♣ Componentes básicos sobre los que debe descansar cualquier tipo de proyecto.
- ♣ Elementos a tener en cuenta para la planificación de un proyecto.
- ♣ Etapas de la gestión de proyectos.

#### **Clase 4.**

- ♣ Viabilidad de un proyecto
- ♣ Ingeniería del software.
- ♣ Visión genérica de la ingeniería del software.

### **Concreción de los programas directores.**

#### **Programa Historia.**

En este caso se hablará sobre el surgimiento histórico de la asignatura, su evolución y sus principales precursores.

#### **Programa Lengua Materna.**

Se harán aclaraciones sobre algunas terminologías que se emplean en la clase y se les dará el tratamiento correspondiente para que los alumnos por sí solos deduzcan su significado.

#### **Programa Salud.**

Enfatizar en las posturas correctas a adoptar cuando se trabaja frente a la PC, importancia de preservar la salud para el correcto cumplimiento de los compromisos adquiridos y mantener un correcto volumen a la hora de trabajar con sonidos, contribuyendo de esta forma a la preservación y cuidado del medio ambiente (**Programa Medio Ambiente**).

#### **Formación de valores.**

Propiciar la valoración por parte de los alumnos del cuidado de los medios tecnológicos con los que trabaja, el cumplimiento de los compromisos asumidos y la colaboración entre los integrantes del equipo de desarrollo y demás.

#### **Intencionalidad Ideopolítica.**

Esta se verá presente durante el transcurso de todas las clases a través del ejemplo del profesor y el cumplimiento de los programas directores.

#### **Métodos.**

Trabajo independiente.

Elaboración conjunta.

**Procedimientos.**

Trabajo independiente.

Elaboración conjunta.

**Medios.**

Pizarra, PC, Libros de textos, Software (PENSOLOG)

**Evaluación en cada una de ellas.**

Preguntas orales y escritas.

Trabajo Práctico.

**Bibliografía.**

Watts S. Humphrey. Introducción al Proceso Software Personal.

Eiriz García, Osana. Metodología para la elaboración del software educativo en la ETP.

Tesis de maestría. Julio/2000.

**Modelación de las actividades:**

**Tema 1:** Introducción a la especialidad, a la disciplina Proyectos Informáticos, a la asignatura Proyectos Informáticos y a los proyectos.

**Clase #2.****Sumario:**

- ♣ Introducción a la disciplina Proyectos Informáticos y a la asignatura Proyectos Informáticos I.
- ♣ Definición de proyecto.
- ♣ Definición de proyecto informático.
- ♣ Características de un proyecto.
- ♣ Ciclo de vida de un proyecto.

**Método:** Elaboración conjunta, método heurístico.

**Medios:** Pizarra, SOFTWARE (PENSOLOG), Fichero ortográfico.

**Bibliografía:**

- ♣ PENSOLOG
- ♣ Documentos digitalizados traídos por el profesor al laboratorio.

**Introducción.**

**(Revisión del estudio independiente)**

La clase se iniciará retroalimentando la clase anterior, teniendo en cuenta la actividad orientada en el estudio independiente, a partir de este momento el profesor aprovecha las potencialidades para hacer alusión al objetivo de la clase.

Precisamente en la clase de hoy vamos a:

**Objetivo**

Caracterizar la disciplina Proyectos Informáticos mediante sus objetos de estudio, sistema de contenidos, evaluación y conceptos básicos contribuyendo a la formación de una conciencia de productor altamente comprometido con su patria.

**Profesor:** En un primer momento el profesor para iniciar el contenido deberá hacer la siguiente pregunta:

**¿Qué se entiende por proyecto?**

**Alumnos:** Se escucharán las definiciones o ideas que expresen los estudiantes, escuchando siempre el criterio individual de ellos. Una vez que ellos den respuestas de una u otra forma se aceptarán sin llegar a nada concreto. Seguidamente:

**Profesor:** Orienta la dirección del **Software (PENSOLOG)** (localhost/PENSOLOG/inicio.exe) como bibliografía de la asignatura.

**Alumnos:** Navegarán en el sistema hasta encontrar las definiciones de proyecto llegando a posibles respuestas.

**Profesor:** Seleccionará a varios estudiantes para darle lectura a los conceptos, extrayendo las palabras esenciales que conforman el concepto. Preguntará las definiciones correctas de las que se muestran en el Sitio.

**Alumnos:** A partir de ese momento tomarán nota de escritura en su libreta de la definición correcta.

**Profesor:** Les dará de manera concreta varios ejemplos donde se pone de manifiesto este concepto.

**Profesor:** en un segundo momento se tendrá en cuenta la siguiente pregunta:

**¿Qué se entiende por proyecto informático?**

**Alumnos:** Se escucharán las definiciones o ideas que expresen los estudiantes, escuchando siempre el criterio individual de ellos. Una vez que ellos den respuestas que de una u otra forma se aceptarán sin llegar a nada concreto. Seguidamente:

**Profesor:** Orienta la dirección del **Software (PENSOLOG)** (localhost/PENSOLOG/inicio.exe) como bibliografía de la asignatura.

**Alumnos:** Navegarán en el sistema hasta encontrar las definiciones de proyecto informático llegando a posibles respuestas.

**Profesor:** Seleccionará a varios estudiantes para darle lectura a los conceptos, extrayendo las palabras esenciales que conforman el concepto. Preguntará las definiciones correctas de las que se muestran en el sitio.

**Alumnos:** A partir de ese momento tomarán nota de escritura en su libreta de la definición correcta.

**Profesor:** Les dará de manera concreta varios ejemplos donde se pone de manifiesto este concepto y hará que el alumno mencione algunos ejemplos de proyecto que quedan excluidos del concepto .

**Profesor:** en un segundo momento se tendrá en cuenta la siguiente pregunta:

**¿Cuáles consideran ustedes que sean las características de un proyecto?**

**Alumnos:** Se escucharán las opiniones o ideas que expresen los estudiantes, escuchando siempre el criterio individual de ellos. Una vez que ellos den respuestas que de una u otra forma se aceptarán sin llegar a nada concreto. Seguidamente:

**Profesor: :** Orienta la dirección del **Software (PENSOLOG)** (localhost/PENSOLOG/inicio.exe) como bibliografía de la asignatura.

**Alumnos:** Navegarán en el sistema hasta encontrar las características de un proyecto llegando a posibles respuestas.

**Profesor:** Seleccionará a varios estudiantes para darle lectura a las opiniones. Preguntará las características correctas de las que se muestran en el sitio.

**Alumnos:** A partir de ese momento tomarán nota de escritura en su libreta de las características de un proyecto.

**Profesor:** Hará que el alumno mencione ejemplos donde se pone de manifiesto estas características.

**Profesor:** en un tercer momento se tendrá en cuenta la siguiente pregunta:

**¿Cuál considera usted que deba ser el ciclo de vida de un proyecto?**

**Alumnos:** Se escucharán las opiniones o ideas que expresen los estudiantes, escuchando siempre el criterio individual de ellos. Se guiará al estudiante, partiendo del desglose de la frase para llegar a conclusiones. Una vez que ellos den respuestas que de una u otra forma se aceptarán sin llegar a nada concreto. Seguidamente:

**Profesor:** Orienta la dirección del **Software (PENSOLOG)** (localhost/PENSOLOG/inicio.exe) como bibliografía de la asignatura.

**Alumnos:** Navegarán en el sistema hasta encontrar el ciclo de vida de un proyecto llegando a posibles respuestas.

**Profesor:** Seleccionará a varios estudiantes para comprender dicha estructura o esquema.

**Alumnos:** A partir de ese momento tomarán nota de escritura en su libreta del ciclo de vida de un proyecto.

**Profesor:** Les dará de manera concreta varios ejemplos donde se pone de manifiesto y hará que el alumno mencione otros.

En todos los momentos donde la participación de los estudiantes sea activa y protagónica se tendrá en cuenta la evaluación de forma cuantitativa, tomando como premisa el criterio valorativo de cada uno de ellos. En el momento indicado se tiene en cuenta los principios de la clase moderna y el trabajo político e ideológico.

**Conclusiones.**

En esta etapa de la clase el profesor hace referencia a los contenidos impartidos, haciendo énfasis en las mayores dificultades ocasionadas por los educandos, citando a los de mayor dificultad en el aprendizaje de acuerdo con las características del grupo.

**Estudio Independiente**

1. Determinar las Fases de un proyecto.
  - b. Explicar alguna de ella.
2. Resuelva los ejercicios 1 y 2 que aparecen en PENSOLOG.
3. ¿Cuáles son las etapas de la gestión de proyectos?
  - a. Explique alguna de ella.

## **Bibliografía**

Sitio Web: alternativa para los alumnos de primer año en los contenidos de la unidad de estudio # 1 de la asignatura de Proyectos Informáticos del Instituto Politécnico de Informática "Carlos Hidalgo Díaz".

**Dirección electrónica:** (localhost/PENSOLOG/inicio.exe)

## **Conclusiones de la actividad.**

Cada clase debe ser ajustada a las características del grupo y en correspondencia con su tipología. En este momento se le pide a algunos profesores ideas de cómo aplicarían este producto en otros tipos de clases.

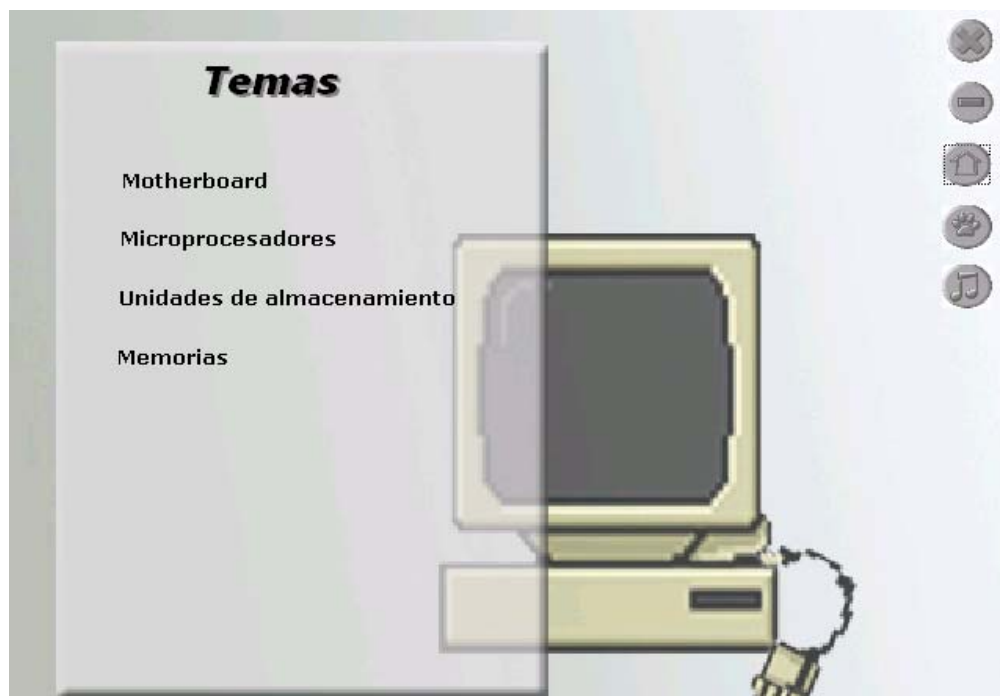


## **Anexo XVI. Manual de usuario.**

**Anexo XVIII. Diseño de las interfaces del resto de las asignaturas.**



**Figura 1. Interfaz para Microsoft Word.**



**Figura 2. Interfaz para Arquitectura.**



Figura 3. Interfaz para Microsoft Access.



Figura 4. Interfaz para Microsoft Excel.



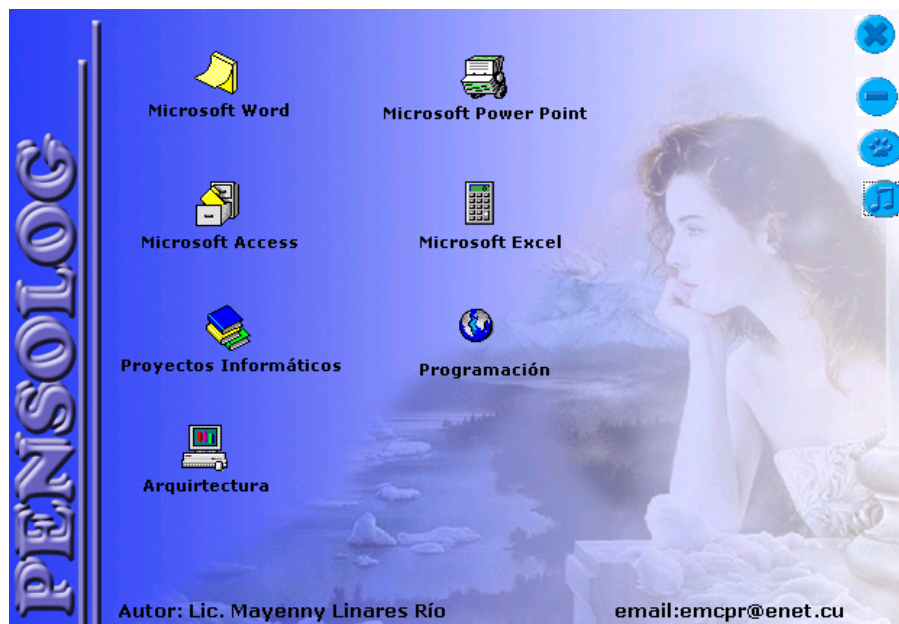
**Figura 5. Interfaz para Lenguaje y Técnica de Programación.**



**Figura 6. Interfaz para Microsoft Power Point.**



**Figura 7. Interfaz para Proyectos Informáticos.**

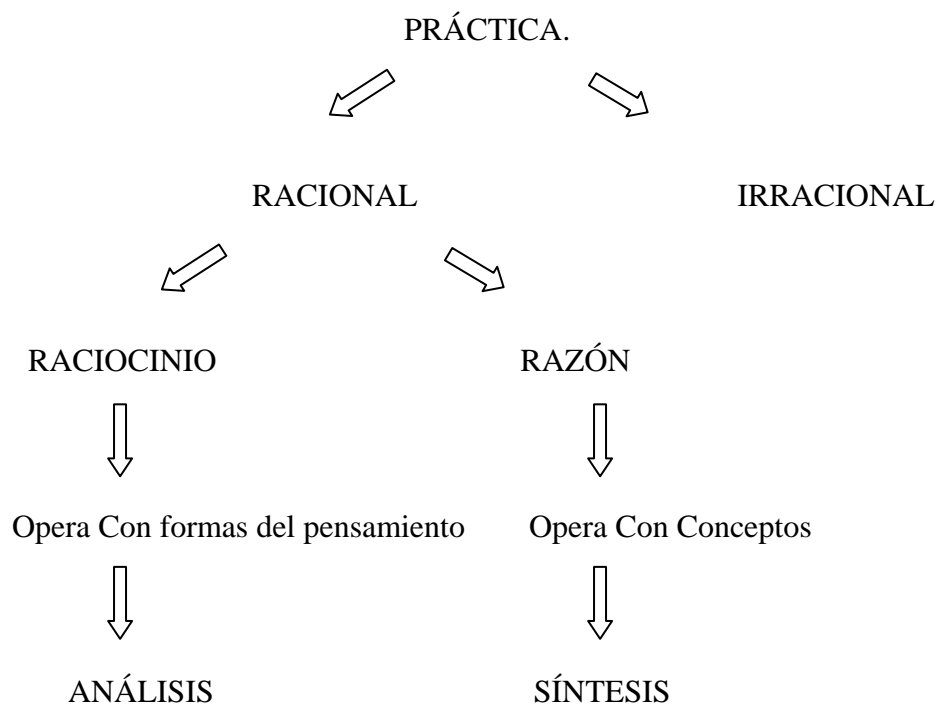
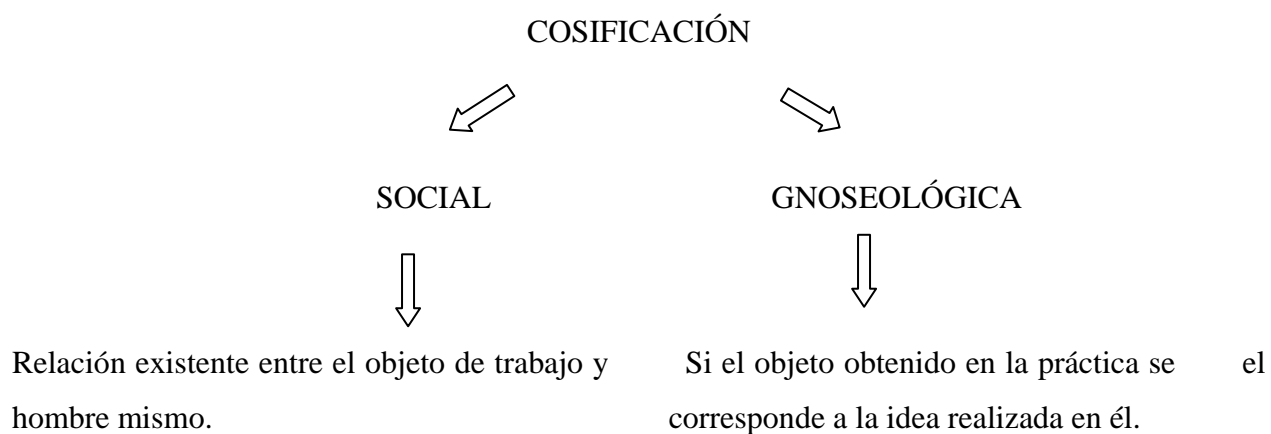


**Figura 8. Interfaz para la selección de asignaturas.**





## Anexo XVIII. Realización práctica de los conocimientos.



**Anexo XIX. Requisitos funcionales.**

Ref. #	Función	Categoría	Atributo	Detalles y restricciones	Categoría.
1.1	Visualizar información.	Evidente	Tiempo de respuesta	5 segundo como máximo	Obligatorio
			Metáfora de interfaz	Pantalla rectangular  Imagen de fondo relacionada con la asignatura  Muestra información con scrooll	Obligatorio
1.2	Presentar ejercicios.	Evidente	Tiempo de respuesta	5 segundo como máximo	Obligatorio
			Metáfora de interfaz	Muestra ejercicio y sus posibles respuestas	Obligatorio
1.3	Promediar resultados.	Ocultá	Tolerancia a fallas	Debe sumar todas las puntuaciones y dividir por la cantidad de ejercicios resueltos cuando no termine de responder la cantidad de ejercicios propuestos.	Obligatorio
1.4	Almacenar datos.	Ocultá	Tolerancia a fallas	Debe almacenar los datos relacionados con los alumnos de cada grupo, aún cuando no se respondan todos los ejercicios ni se llenen todos los datos.	Obligatorio
1.5	Mostrar datos según la selección.	Evidente	Tiempo de respuesta	5 segundo como máximo	Obligatorio
			Metáfora de interfaz	Pantalla rectangular  Degradado azul-blanco de fondo  Organización lógica de los datos  Mostrar botones	Obligatorio
			Tolerancia a fallas	Mostrará solo los datos que adecuen a las respuestas dadas por el alumno.	Obligatorio
1.6	Enviar mensajes.	Superflua	Tiempo de respuesta	5 segundo como máximo	Obligatorio

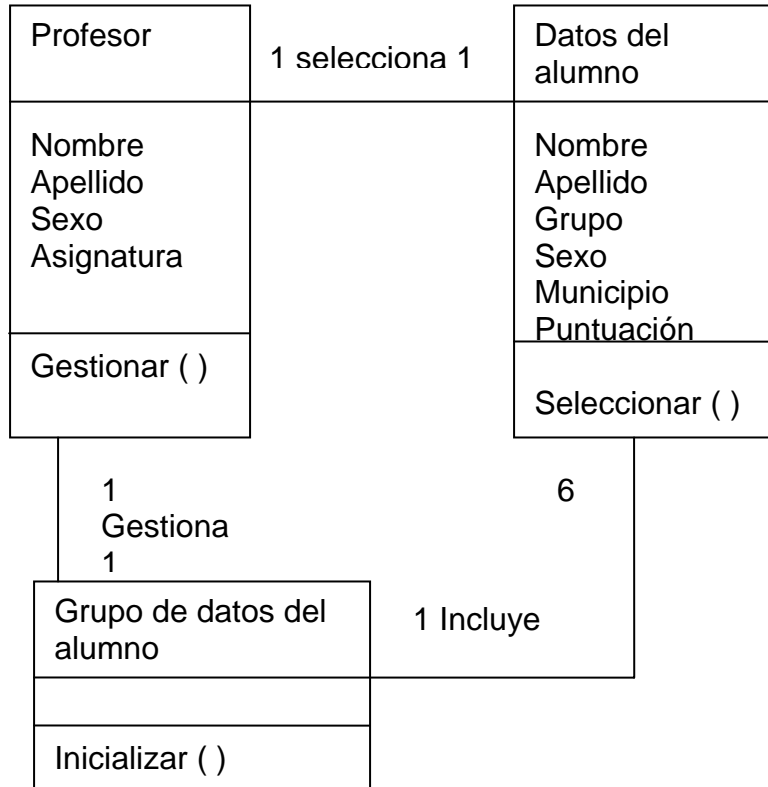


**Anexo XX. Casos de uso.**

CASO DE USO DEL SOFTWARE		Gestión del alumno por parte del profesor.
Actores	Profesor (Inicia)	
Propósito	Eliminar, transformar o adicionar los datos de cada alumno, partiendo de la visualización y manipulación de la base de datos.	
Resumen:		
El caso de uso se inicia cuando el profesor accede al software, se identifica y selecciona el acceso a la gestión de alumnos como opción. En este caso aplicará las transformaciones a realizar.		
Curso normal de los eventos.		
Acción del actor		Respuestas del sistema.
1. El caso de uso comienza cuando el usuario ejecuta el software.		2. Se identifica
3. El usuario se identifica.		4-a. Se almacenan los datos. 4-b. Se presenta la ventana principal con las opciones a realizar
5. El usuario escoge la opción de Gestionar alumnos		6. Muestra la base de datos
7. Aplica transformaciones.		8. Muestra la información
9. Salir del software		10. Cierra el programa
Prioridad		Responde al principal objetivo de automatización al resolver parte del problema.
Mejoras		Datos que contiene el producto
Otras secciones		



## Diagrama de clases.

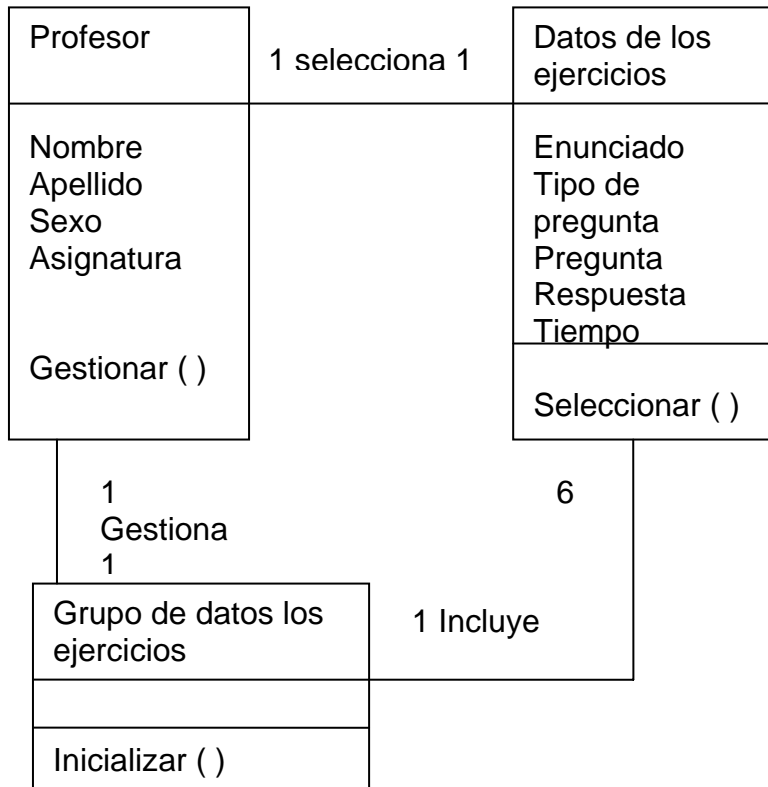




CASO DE USO DEL SOFTWARE		Gestión de los ejercicios por parte del profesor.
Actores	Profesor (Inicia)	
Propósito	Eliminar, transformar o adicionar los ejercicios, partiendo de la visualización y manipulación de la base de datos.	
<b>Resumen:</b> El caso de uso se inicia cuando el profesor accede al software, se identifica y selecciona la gestión de ejercicios como opción. En este caso aplicará las transformaciones deseadas para visualizar los datos necesarios.		
Curso normal de los eventos.		
Acción del actor		Respuestas del sistema.
1. El caso de uso comienza cuando el usuario ejecuta el software.		2. Se identifica
3. El usuario se identifica.		4-a. Se almacenan los datos. 4-b. Se presenta la ventana principal con las opciones a realizar
5. El usuario escoge la opción de Gestionar ejercicios		6. Muestra pantalla de explicación de los pasos a dar
7. Aplica los pasos.		8. Muestra la información a transformada.
9. Salir del software		10. Cierra el programa
Prioridad		Responde al principal objetivo de automatización al resolver parte del problema.
Mejoras		Datos que contiene el producto
Otras secciones		



### Diagrama de clases.





## Anexo XXI. Imágenes de la aplicación de herramientas en el software.

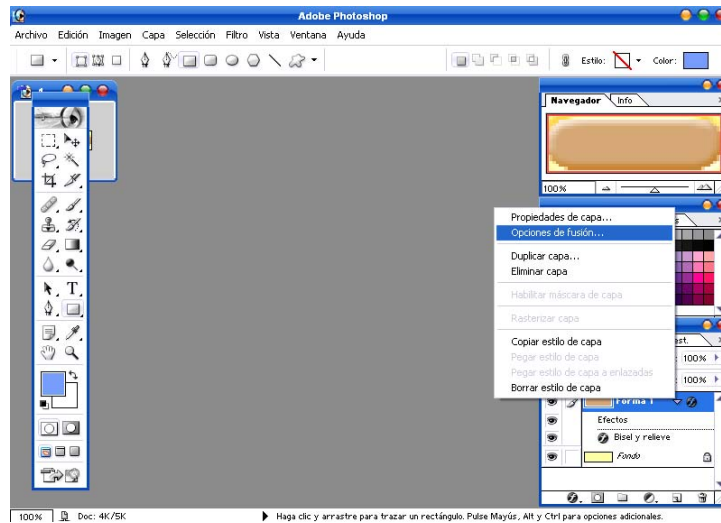


Figura 1. Propiedades de fusión.

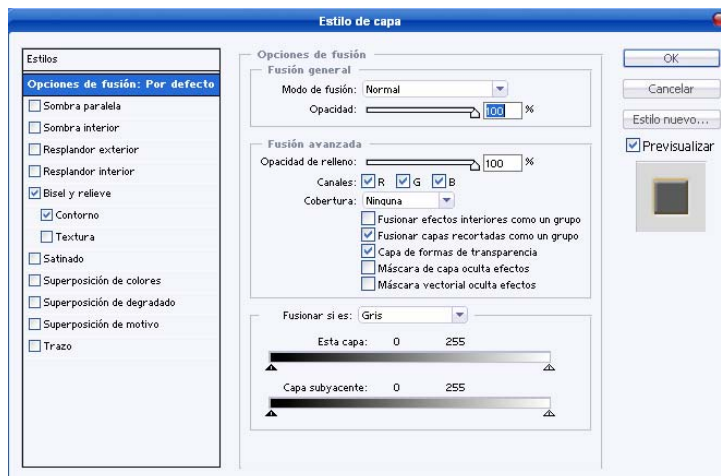


Figura 2. Propiedades de interfaz.

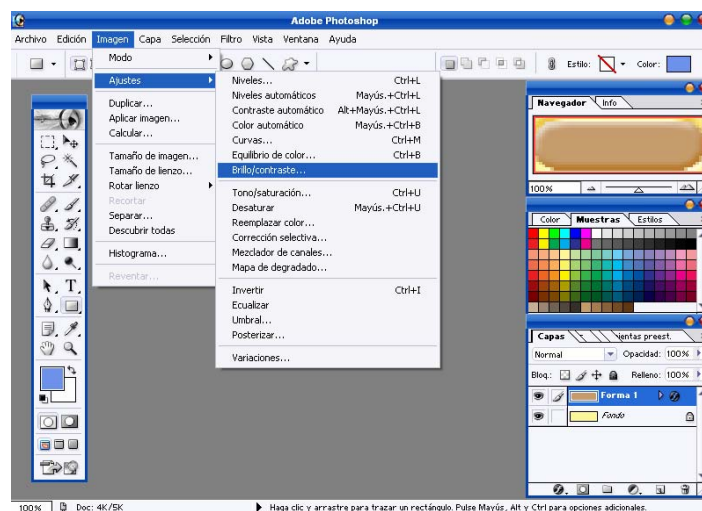


Figura 3. Ajustes de imágenes.

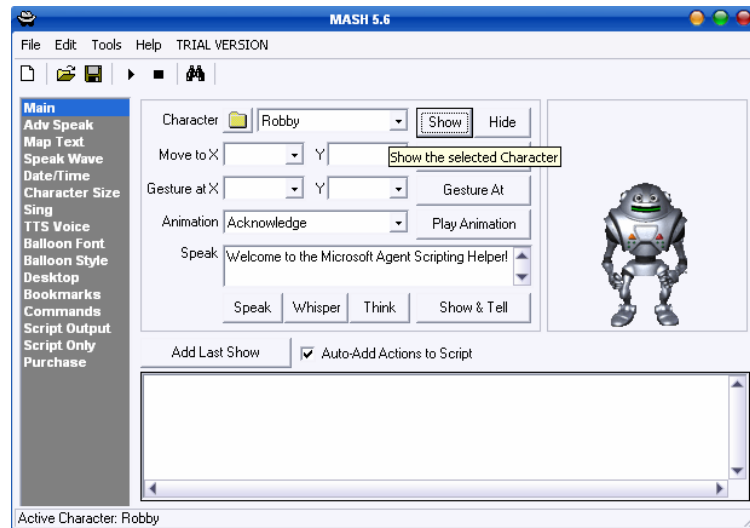


Figura 4. Selección de mascota.

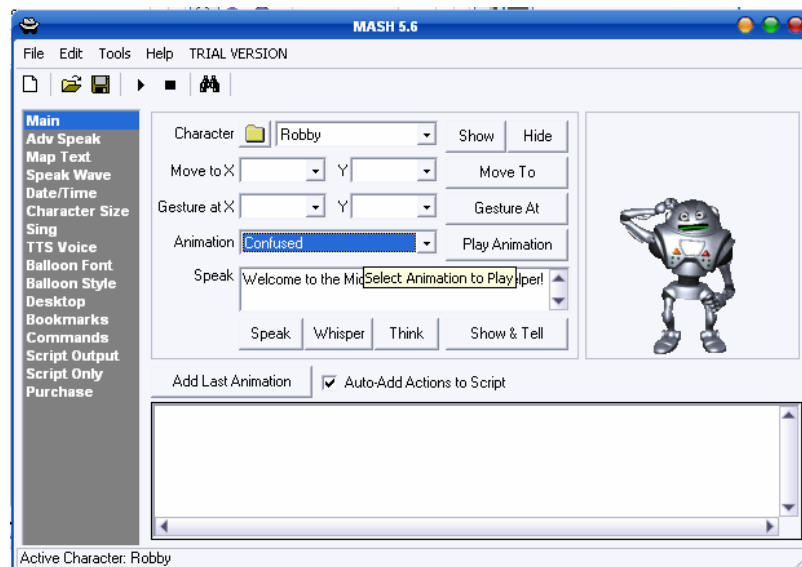


Figura 5. Buscar la acción a ejecutar.

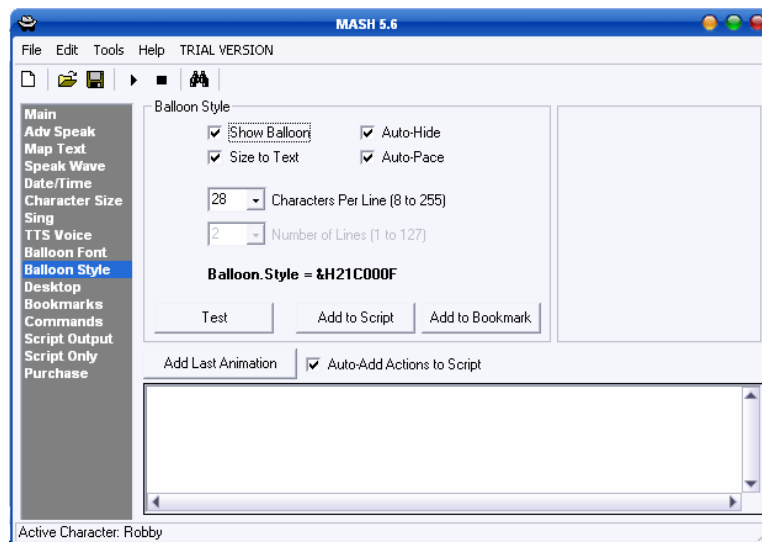
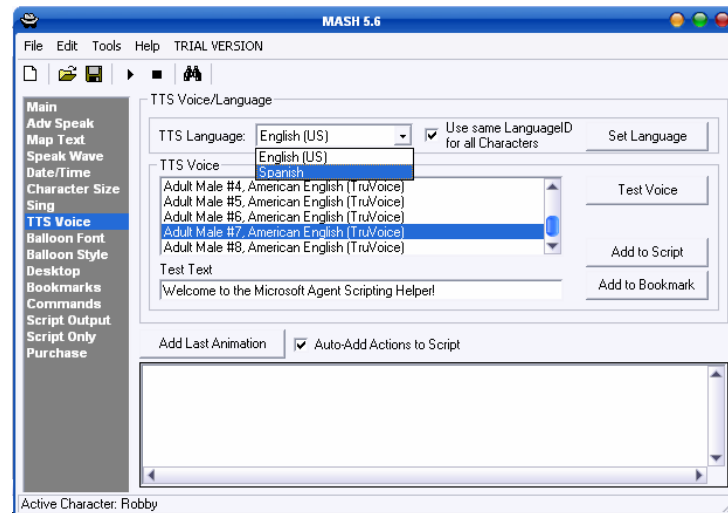


Figura 6 y 7. Configuración de lenguaje y balones.

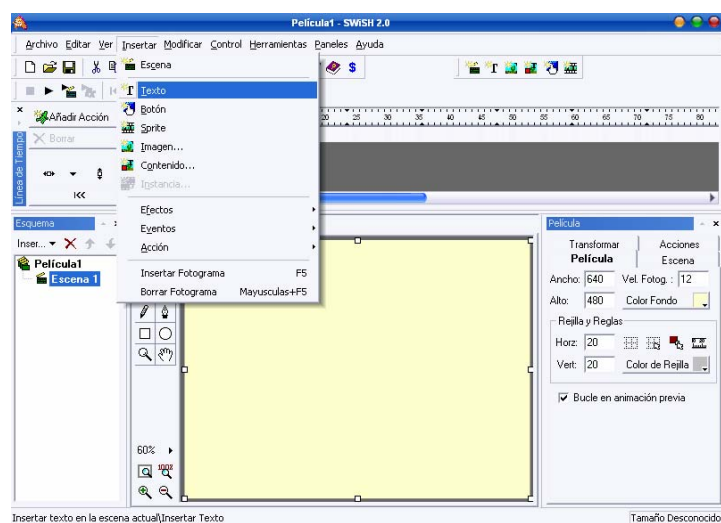


Figura 8. Inserción de textos e imágenes.



## Anexo XXII. Codificaciones utilizadas en el recurso.

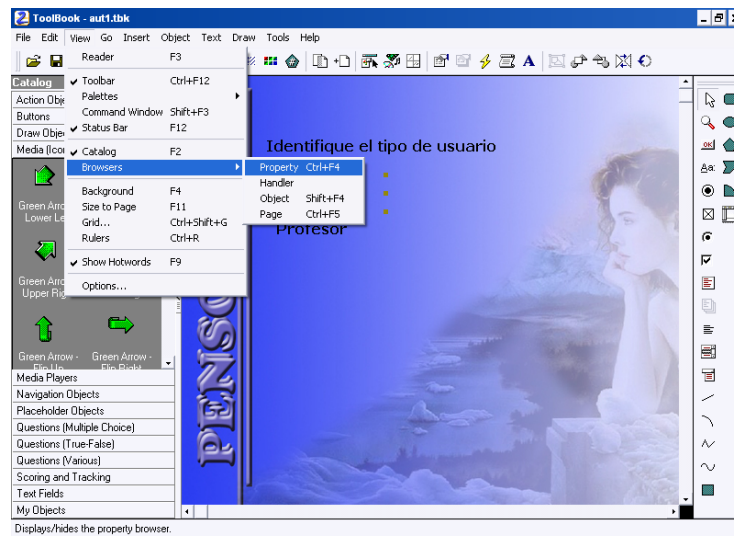


Figura 2. Camino a seguir para dar propiedades específicas a los objetos.

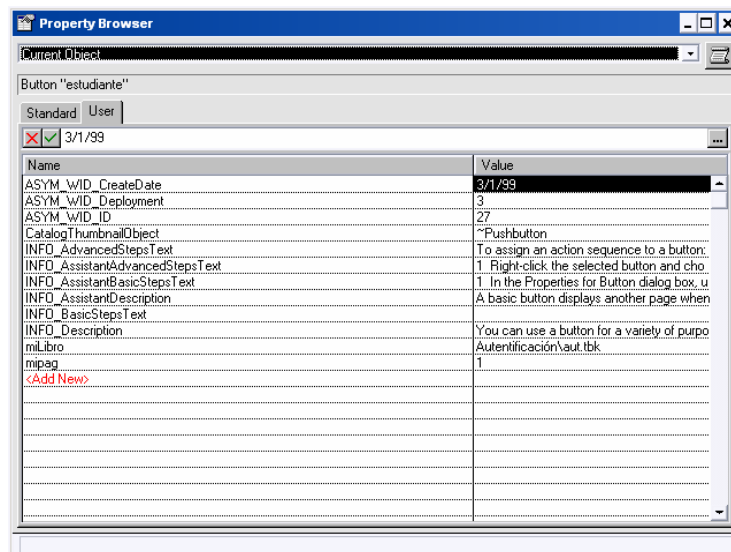


Figura 3. Pantalla de propiedades del objeto.

```
to handle buttonclick
  go to page ProfeEstudiante
end buttonclick
```

Figura 4. Código para ir a la pantalla anterior y cambiar de usuario.



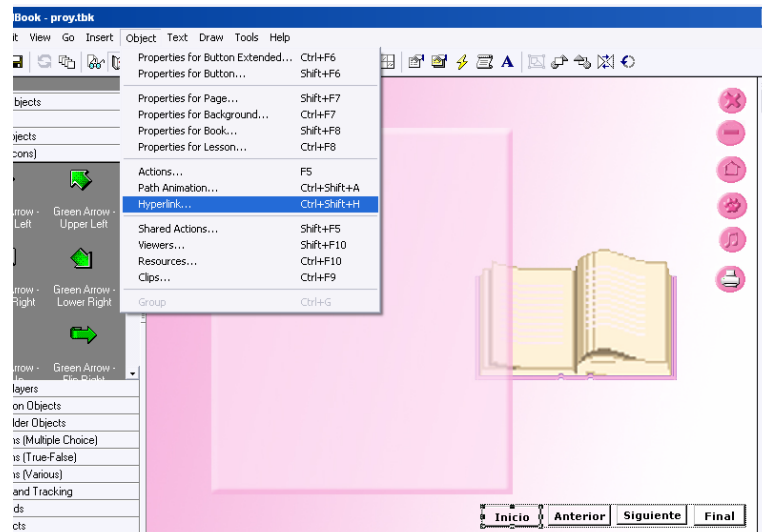


Figura 5. Asociación de hipervínculos a objetos.

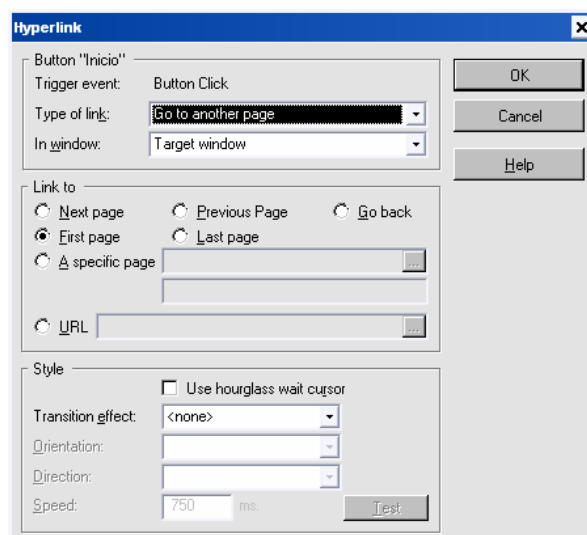


Figura 6. Propiedades del hipervínculo.

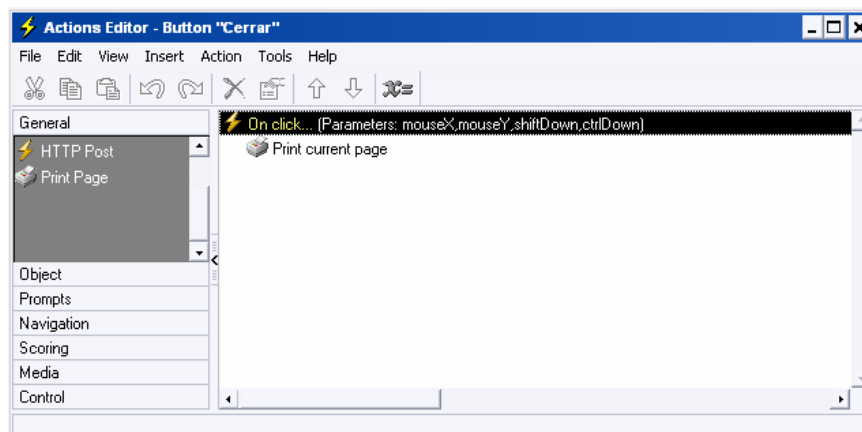


Figura 7. Asociación de la acción imprimir a través del editor de acciones.



```
to handle buttonclick
  sysLockScreen=True
  If bounds of target=migrande of target
    stretchgraphic of target=true
    bounds of target=michiqui of target
    syscursor=17
  else
    stretchgraphic of target=false
    bounds of target=migrande of target
    syscursor=18
  end if
  sysLockScreen=False
end
```

**Figura 8. Código para ver imagen agrandada.**

```
to handle ButtonClick
  system CaminoExe
  system Maestro
  System Titulo, PieFoto, Opcion

  If (KeyState(KeyControl) = "Down") And not(sysRuntime) then
    -- Entrada en modo de Edición
    Titulo = Text of Field "Nombrefoto" of Parent of target
    PieFoto = PieFoto of Target
    Send EditarMiniatura Self
```

**Figura 9. Código para la ejecución de videos.**

```
to handle ButtonClick
  send Sort
end
```

**Figura 10. Código para ordenar datos.**

```
To handle ButtonClick
  Send PrintReport
End
```

**Figura 11. Código para hacer informes.**



```
to handle ButtonClick
Request "Está seguro que desea eliminar este artículo" with Sí or No
if IT is Sí
send SelectPage
send Clear
end
end
NotifyAfter EnterPage
if PageCount of this Book = 1
enabled of self = false
else
enabled of self = True
end if
end
```

**Figura 12. Código para dar alta a un estudiante.**

```
to handle ButtonClick
send NewPage
end
```

**Figura 13. Código para búsqueda de un estudiante.**