

**DISEÑO INSTRUCCIONAL BASADO EN COMPETENCIAS Y
CONSTRUCCIÓN DE UN OBJETO DE APRENDIZAJE RELACIONADO
CON UNATEMÁTICA DE LA ASIGNATURA PROGRAMACIÓN
ORIENTADA A OBJETOS DEL PROGRAMA ACADÉMICO DE
INGENIERÍA DE SISTEMAS E INFORMÁTICA**

**LUZ NELLY MARTÍNEZ AYALA
HELMER LEONEL CALDERÓN ROJAS**

**UNIVERSIDAD INDUSTRIAL DE SANTANDER
FACULTAD DE INGENIERÍAS FÍSICO-MECÁNICAS
ESCUELA DE INGENIERÍA DE SISTEMAS E INFORMÁTICA
BUCARAMANGA
2008**

**DISEÑO INSTRUCCIONAL BASADO EN COMPETENCIAS Y
CONSTRUCCIÓN DE UN OBJETO DE APRENDIZAJE RELACIONADO
CON UNA TEMÀTICA DE LA ASIGNATURA PROGRAMACIÓN
ORIENTADA A OBJETOS DEL PROGRAMA ACADÉMICO DE
INGENIERÍA DE SISTEMAS E INFORMÀTICA**

**TRABAJO DE GRADO PARA OPTAR AL TÍTULO DE: INGENIERO(A) DE
SISTEMAS**

**AUTORES: LUZ NELLY MARTÍNEZ AYALA
HELMER LEONEL CALDERÓN ROJAS**

**DIRECTOR: JORGE HERRERA CASTILLO
Ingeniero de Sistemas**

**CODIRECTOR: CLARA INES PEÑA DE CARRILLO
Directora Científica CENTIC**

**UNIVERSIDAD INDUSTRIAL DE SANTANDER
FACULTAD DE INGENIERÍAS FÍSICO-MECÁNICAS
ESCUELA DE INGENIERÍA DE SISTEMAS E INFORMÁTICA
BUCARAMANGA
2008**

AGRADECIMIENTOS

Los autores expresan sus agradecimientos a:

Al Director del Proyecto, **Jorge Herrera Castillo**, por su confianza, apoyo y aportes realizados para culminar este proyecto.

A la codirectora **Clara Inés Peña de Carrillo**, directora científica del CENTIC, por su motivación y dedicación al desarrollo de los proyectos Prospetic.

Al grupo de investigación y desarrollo del CENTIC por la colaboración y el soporte técnico al proyecto.

A nuestras familias por el apoyo incondicional.

A nuestros amigos, quienes estuvieron incondicionalmente motivándonos.

A la Universidad Industrial de Santander, por habernos permitido crecer de manera integral a nivel personal y profesional.

A todas las personas que de una u otra forma, colaboraron o participaron en el desarrollo de este trabajo de grado.

DEDICATORIA

*A Dios por ser mi mejor amigo, quien me dio la fe,
la fortaleza, la salud y la esperanza para
terminar este trabajo.*

*A ti madre, fiel amiga, acompañante y consejera, a tu
sacrificio debo lo que soy y he querido ser. Te agradezco
las lágrimas derramadas, el apoyo incondicional y
la espera a que este momento llegara. Hoy me considero
una mujer con valores éticos y sociales, es porque los
heredé de ti. Gracias, mami, por cuidarme y en cada
momento hacerme sentir especial y por recordarme
que no podía darme el lujo de desfallecer”.*

*A mi hermana, Sandra Patricia, por tu afecto e
inolvidables momentos compartidos.*

*A esos dos ángeles que me regalan sus pícaras sonrisas que
iluminan mi existencia, María Alejandra y Daniela.*

*Y a Ti por haber llegado a mi vida y ser apoyo en este
camino recorrido.*

NEILY FORTALEZA

DEDICATORIA

*Gracias infinitas
A dios, por el camino recorrido
A mis padres, por su amor y apoyo
A la vida por lo aprendido y
A todas las personas que colaboraron
de manera directa o indirecta
para que todo saliera bien.*

Leonel Calderón

CONTENIDO

INTRODUCCIÓN	20
1. ASPECTOS GENERALES	21
1.1 PLANTEAMIENTO DEL PROBLEMA	21
1.2. OBJETIVOS	22
1.1.1 Objetivo General	22
1.1.2 Objetivos Específicos	22
1.3 JUSTIFICACIÓN	23
2. MARCO TEÓRICO	25
2.1 DISEÑO INSTRUCCIONAL	25
2.1.1 Formación basada en competencias	26
2.1.2 Teorías educativas	27
2.1.3 Estilo aprendizaje	27
2.1.4 Tecnologías de la información y comunicación (TIC´s) en la educación	30
2.1.5 Fundamentos de E-learning	32
2.1.6 El análisis funcional	33
2.2 ETAPAS DE CONSTRUCCION Y DESARROLLO DE LAPROUESTA	34
2.2.1 Fase 1. Definición	34
2.2.2 Fase 2, Diseño instruccional	35
2.2.3 Fase 3, Diseño y construcción de objeto y aprendizaje	36
2.2.4 Fase 4, Integración y evaluación en la plataforma e-escen@riUIS	36
2.2.5 Fase 5, Puesta en marcha	36
2.2.6 Fase 6, Conclusiones y cierre	36
2.2.7 Fase 7, Seguimiento y control de calidad	37
2.3 HERRAMIENTAS PARA EL DESARROLLO DEL SISTEMA	37
2.3.1 Hardware	37
2.3.2 Software	37
3. APLICACIÓN DE LA PROUESTA METODOLOGICA AL DISEÑO INSTRUCCIONAL BASADO EN COMPETENCIAS DE LA ASIGNATURA PROGRAMACION ORIENTADA A OBJETOS	39
3.1. ETAPAS DE CONSTRUCCIÓN Y DESARROLLO DEL DISEÑO INSTRUCCIONAL	39
3.3.1 Diagrama secuencial de actividades de aprendizaje DSA ²	39
3.3.2 Planteamiento de los saberes y haceres relacionados	45
3.3.3 Estructuración Modular	47
3.3.4 Actividades propósitos de formación	50
3.3.5 Planeación Curricular	51
4. METODOLOGÍA PARA LA GENERACIÓN Y ENCAPSULAMIENTO DEL OBJETO DE APRENDIZAJE	58
4.1 DEFINICIÓN Y CARACTERÍSTICAS DEL OBJETO DE APRENDIZAJE	58
4.1.1 Nombre del Objeto de Aprendizaje	60
4.1.2 Objetivo del Objeto de Aprendizaje	60
4.1.3 Contenido del Objeto de Aprendizaje	60
4.1.4 Conexión entre el Objeto de Aprendizaje y el Diseño Instruccional	60
4.1.5 Sobre la Aplicación del Objeto de Aprendizaje	60
4.1.6. Sobre la Evaluación del Objeto de Aprendizaje	61
4.1.7 Sobre los Vínculos de Profundización del Contenido	61
4.1.8 Declaración de Autoría del Contenido	61
4.2 PROCESO DE GENERACIÓN DE UN OBJETO DE APRENDIZAJE	61

5. DISEÑO Y DESARROLLO DE OBJETOS DE APRENDIZAJE QUE SOPORTAN LA TEMATICA CONCEPTOS DE CLASES Y OBJETOS	63
5.1 CONCEPTO Y CARACTERÍSTICAS DEL OBJETO DE APRENDIZAJE	63
5.1.1 Nombre del Objeto de Aprendizaje	64
5.1.2 Objetivos del Objeto de Aprendizaje	64
5.1.3 Contenido del Objeto de Aprendizaje	64
5.1.4 Diseño del applet	76
5.1.5 Lenguaje utilizado del producto software	76
5.1.6 Instrucciones utilizadas	77
5.1.7 Evaluación del objeto de aprendizaje	80
5.2 ENCAPSULAMIENTO DEL OBJETO DE APRENDIZAJE	86
5.2.1 Generación de metadatos y encapsulamiento del objeto	86
6. PORTAL DEL PROFESOR	94
CONCLUSIONES	97
RECOMENDACIONES	98
BIBLIOGRAFIA	99
ANEXOS	102

LISTA DE TABLAS

Tabla 1	Estrategia Instruccional. Componentes de un curso hipermedia para los objetos de aprendizaje de una unidad docente en e-escen@riUIS	29
Tabla 2	Materiales Instruccionales Complementarios y Elementos de Interactividad y de Evaluación.	29
Tabla 3	Formato del Material	30
Tabla 4	Herramientas de Navegación	30
Tabla 5	Fases y productos del diseño curricular basado en competencias	35
Tabla 6	Estrategias y técnicas de enseñanza/aprendizaje	55
Tabla 7	Técnicas e instrumentos de evaluación	56

LISTA DE FIGURAS

Figura 1	Ingeniería Instruccional	25
Figura 2	Fases del proyecto	34
Figura 3	Fases de la construcción del Diseño Instruccional	39
Figura 4	Programa Académico niveles 1 y 2 de la escuela de Ingeniería de sistemas e informática	40
Figura 5	Objetivo de la asignatura	41
Figura 6	Esquema general del DSA ²	41
Figura 7	Convenciones empleadas DSA ²	42
Figura 8	Preconceptos en el DSA ²	42
Figura 9	Dependencia en el DSA ²	43
Figura 10	Transversalidad en el DSA ²	44
Figura 11	Causa-consecuencia en el DSA ²	44
Figura 12	Paralelismo en el DSA ²	45
Figura 13	Estructura gramatical de los Saberes	46
Figura 14	Estructura gramatical de los contenidos desagregados	46
Figura 15	Tabla de saberes	47
Figura 16	Módulos de formación de los conocimientos de la programación	48
Figura 17	Unidad de aprendizaje pilares de la POO	48
Figura 18	Actividad de formación Definición y Objetivos de la POO	49
Figura 19	Niveles de la estructuración modular para el primer módulo de la asignatura POO	50
Figura 20	Tabla propósitos-actividades	51
Figura 21	Elementos de la Planeación Curricular	52
Figura 22	Módulo de formación de la Planeación Curricular	52
Figura 23	Escenarios y duración del módulo de formación de la Planeación Curricular	52
Figura 24	Propósito y metodología de enseñanza-aprendizaje del módulo de formación de la Planeación Curricular	53
Figura 25	Técnicas e instrumentos de evaluación del módulo de formación de la Planeación Curricular	54
Figura 26	Medios didácticos para el objeto de aprendizaje del primer módulo de la planeación curricular	57
Figura 27	Estructura de un objeto de aprendizaje	59
Figura 28	Tabla de contenido del Objeto de la asignatura POO	65
Figura 29	Núcleo de Conocimiento	66
Figura 30	Explicación de la plantilla Web para el objeto de aprendizaje	66
Figura 31	Explicación de los elementos adicionales para el objeto de aprendizaje	68

Figura 32	Material en la plantilla para el subtema Introducción	69
Figura 33	Material en la plantilla para el subtema Clases y Objetos en el contexto de diseño y programación	70
Figura 34	Material en la plantilla par el subtema Los Métodos	71
Figura 35	Material en la plantilla para el subtema La Visibilidad	72
Figura 36	Material en la plantilla para el subtema El Paso de Mensajes	73
Figura 37	Material en la plantilla para el subtema Eventos	74
Figura 38	Material en la plantilla para el subtema Diseño y programación de aplicaciones orientado a objetos	75
Figura 39	Interfaz del Objeto de Aprendizaje	77
Figura 40	Atributos y métodos de los objetos	78
Figura 41	Espacio de memoria para el objeto	79
Figura 42	Implementación de la clase y el objeto en el lenguaje java	79
Figura 43	Diagrama de clases UML	80
Figura 44	Guía para el manejo del objeto de aprendizaje	80
Figura 45	Escritorio de la plataforma e-escen@riUIS	81
Figura 46	Ventana para la gestión de la evaluación	82
Figura 47	Ventana para la gestión de ejercicios	83
Figura 48	Ventana para la gestión de la evaluación	84
Figura 49	Ejercicios construidos en la temática concepto de clases y objetos	85
Figura 50	Actividades de Trabajo colaborativo	85
Figura 51	Creación de un paquete SCORM	87
Figura 52	Escritorio de trabajo de la herramienta RELOAD	88
Figura 53	Edición de los metadatos	89
Figura 54	Introducción en un LMS	90
Figura 55	Creación de la carpeta metadatos	91
Figura 56	Añadir la Organización a la Estructura del Objeto de Aprendizaje	92
Figura 57	Creación de un paquete con RELOAD	93
Figura 58	Página principal del portal del profesor	94
Figura 59	Enlaces a Docencia	95

LISTA DE ANEXOS

ANEXO A	Cuestionario de estilos de aprendizaje según el modelo de Felder y Silverman	103
ANEXO B	Contenidos temáticos generales de la asignatura POO	109
ANEXO C	DSA ²	113
ANEXO D	Planteamiento de los saberes y haceres relacionados	114
ANEXO E	Establecimiento de la relación propósitos-actividades de formación	118
ANEXO F	Estructuración Modular	121
ANEXO G	Planeación Curricular	124

GLOSARIO

Análisis Funcional: Es una metodología de investigación que permite identificar, luego de desarrollar una serie de etapas, las competencias inherentes al ejercicio de una función productiva.

Aproa: Aprendiendo con Repositorios de Objetos de Aprendizaje. El proyecto APROA presenta un paquete tecnológico basado en el uso de tecnologías de Objetos de Aprendizaje, los cuales se han convertido en una potente opción para potenciar las capacidades docentes de las instituciones de educación superior.

Causa-consecuencia: Evidencia que existe información necesaria y suficiente entre el tema origen y el tema de destino involucrados en el proceso de aprendizaje.

Competencias: Es una construcción social compuesta de aprendizajes significativos en donde se combinan atributos tales como conocimientos, actitudes, valores y habilidades.

Compilar: Traducir un lenguaje de programación de alto nivel, como C en el lenguaje de ensamblador para preparar la ejecución.

C Sharp: lenguaje de programación orientado a objetos desarrollado y estandarizado por Microsoft como parte de su plataforma NET.

Dependencia: Permite que dos temas se contextualicen en el proceso de aprendizaje de la asignatura Programación Orientada a Objetos.

Diagrama Secuencial de Actividades de Aprendizaje: Es la forma como se estructura la materia, se asocian sus temáticas para el proceso de aprendizaje de la asignatura.

Diseño Curricular: Diseño de un plan para facilitar el aprendizaje y el logro de metas y objetivos relacionados con una unidad escolar específica, contempla medios y objetivos, así como estrategias de instrucción y evaluación. Una propuesta curricular convencional se diseña simplemente en torno a contenidos, objetivos y evaluación; su integración puede ser clara a la hora de su desarrollo y nunca indica cómo llegar al logro del egresado competente. Una propuesta curricular por competencias se diseña entonces en torno a los perfiles profesionales esperados y se hace cargo de las necesidades sociales; las competencias definen los criterios para la selección y organización de todas las componentes y ofrece metodologías para la evaluación de las competencias de desempeño.

Diseño Instruccional: Disciplina que aplica una metodología basada en la teoría instruccional para impartir y crear contenidos formativos.

E-escen@ri_{uis}: Plataforma educativa institucional de la UIS, denominada escenario electrónico de recursos de aprendizaje e investigación.

E-learning: Palabra que denota el uso de las tecnologías multimedia para desarrollar y mejorar nuevas estrategias de aprendizaje.

Estilos de Aprendizaje: Son los rasgos cognitivos, afectivos y fisiológicos que sirven como indicadores relativamente estables, de cómo los alumnos perciben interacciones y responden a sus ambientes de aprendizaje.

Estrategias: Es una guía que orienta la obtención de ciertos resultados, los cuales deben tener los métodos didácticos que mejor se adaptan al tipo de razonamiento identificado es decir la estrategia debe ser flexible.

Estructuración modular: La estructuración modular se logra a partir de los propósitos identificados para la asignatura y la tabla de saberes y haceres; debe ser secuencial es decir, que se agrupan por afinidad propósitos y saberes, identificando de esta forma acciones delimitadas y manteniendo la relación de causa-consecuencia entre las diferentes desagregaciones.

Evidencia de aprendizaje: Son los referentes que permiten la asimilación del aprendizaje del estudiante, o las acciones demostrables que debe realizar el estudiante para corroborar ante el mismo y ante el proceso de enseñanza, el aprendizaje.

Java: lenguaje de programación orientado a objetos de licencia libre, desarrollado por Sun Microsystems.

Metadatos: Son datos acerca de los datos, proveen un mecanismo para describir el contenido de un asset, un SCO o un objeto de aprendizaje, además de encontrarlo dentro de un repositorio. Esta lista de metadatos se divide en 9 categorías con un total de 62 campos entre todos; algunos de estos campos utilizan un vocabulario pre-definido.

Módulo de Formación: Es el núcleo de la estructura curricular asociado a la unidad de competencia, integrado por unidades de aprendizaje; autosuficiente, de uso flexible de acuerdo con las necesidades específicas de formación de objeto certificación.

Objeto de Aprendizaje: El concepto utilizado en los proyectos institucionales adoptado para el proyecto ProSPETIC es: Un objeto de aprendizaje es una entidad digital basada en la aplicación de la metodología del análisis funcional para programas de formación por competencias (diseño instruccional) que puede ser utilizado, reutilizado o referenciado durante el

aprendizaje en línea con el objetivo de generar conocimientos, habilidades y destrezas en función de las necesidades del estudiante.

Paralelismo: Los temas que se desagregan del tema origen poseen el mismo grado de importancia y por tanto pueden ser abordados en cualquier orden en el proceso de aprendizaje.

Planeación curricular: La planeación curricular constituye un proceso fundamental en el desarrollo de esta propuesta metodológica, pues a través de ella se consolida el diseño instruccional de la asignatura.

Preconcepto: Evidencia que existe información necesaria aunque no suficiente para abordar el tema por lo tanto se requiere información adicional que permita el proceso de enseñanza-aprendizaje.

Procesador: Componente lógico de un sistema de computación que interpreta y ejecuta instrucciones de programas.

Programación Orientada a Objetos: Técnica o estilo de programación que utiliza objetos como bloque esencial de construcción.

Reload: Es una herramienta para crear y editar paquetes e insertar metadatos conforme a las especificaciones de ADL² e IMS³.

SCO: SCO (Objeto de Contenido compartible). Objeto de aprendizaje distribuable, mínima expresión de contenido formativo con entidad por sí mismo, etiquetado con metadata para permitir su búsqueda y recuperación, y que puede ser agregado a otras SCOs para crear unidades de instrucción de mayor entidad. Se habla de SCO para cualquier objeto de aprendizaje (LO) que implemente la especificación SCORM.

SCORM: Modelo de Referencia para Objetos de Contenido Distribuibles. SCORM es un modelo de referencia que establece un modo de desarrollar, empaquetar y gestionar la distribución de unidades formativas digitales (reusable, accesible, interoperable, duradero).

Software: Programa, equipamiento lógico o soporte lógico a todos los componentes intangibles de una computadora, es decir, al conjunto de programas y procedimientos necesarios para hacer posible la realización de una tarea específica.

Taxonomía de Bloom: Se define como la idea de asir el significado de elementos o de las cosas. La taxonomía de Bloom divide en tres dominios la forma en que las personas aprenden y propone seis niveles de competencia de los objetivos formativos en el diseño curricular.

Transversalidad: Es un tema que se requiere para múltiples temas en diferentes espacios de tiempo y contextos para el proceso de aprendizaje (se desea evitar la redundancia de temas dentro de la asignatura).

Tabla de Saberes: Instrumento que permite precisar y diferenciar los saberes que están integrados en una unidad de aprendizaje: conceptos, principios y teorías, procedimientos cognitivos y motrices, actitudes y valores contemplados para el desarrollo de la unidad de aprendizaje.

Unidad de Aprendizaje: Referente técnico pedagógico que permite la organización del trabajo del instructor para la orientación del proceso de aprendizaje, bien sea en aulas, talleres, laboratorios, empresas, comunidades y otros entornos de formación.

RESUMEN

TÍTULO*:

DISEÑO INSTRUCCIONAL BASADO EN COMPETENCIAS Y CONSTRUCCIÓN DE UN OBJETO DE APRENDIZAJE RELACIONADO CON UNA TEMÁTICA DE LA ASIGNATURA: PROGRAMACIÓN ORIENTADA A OBJETOS DEL PROGRAMA ACADÉMICO DE INGENIERÍA DE SISTEMAS E INFORMÁTICA.

AUTORES:**

LUZ NELLY MARTÍNEZ AYALA, HELMER LEONEL CALDERÓN ROJAS.

PALABRAS CLAVES:

Análisis funcional, diseño instruccional, planeación curricular, objetos de aprendizaje, estilos de aprendizaje, programación orientada a objetos.

DESCRIPCIÓN:

Actualmente la educación se basa en procesos de continua evolución que ofrecen estrategias metodológicas enfocadas al uso de herramientas vigentes y variadas; por esta razón, la educación se está orientando hacia el desarrollo de competencias en el estudiante; para tal fin se realizan cambios o revisiones en los diseños instruccionales, con el propósito de detectar y desarrollar en el alumno habilidades, destrezas, conocimientos y competencias que le permitan adecuarse al cambio actual de la sociedad.

Este documento describe la manera en la cual se desarrolló el Diseño Instruccional basado en competencias de la asignatura Programación Orientada a Objetos, para una posterior aplicación de las Tecnologías de la información y comunicación (TIC's). El desarrollo de esta propuesta metodológica orientada a la formación basada en competencias elaborada mediante el planteamiento de 5 etapas: Análisis de actividades de aprendizaje, Planteamiento general de saberes, Establecimiento de la relación de actividades-propósitos, Estructuración curricular, de la cual hacen parte la identificación de las actividades de formación, estructuración de las unidades de aprendizaje y la identificación de los módulos de formación. Finalmente se elabora la planeación curricular, conformada por los criterios, los contenidos, las estrategias y técnicas de aprendizaje (basadas en los estilos de aprendizaje de Felder y Silverman), las evidencias de aprendizaje, las técnicas y los instrumentos de evaluación, la duración, los recursos y escenarios, además de la elaboración de un objeto de aprendizaje que permita dar soporte al proceso de enseñanza –a aprendizaje relacionado con la temática “Concepto de Clases y Objetos”.

* Proyecto de grado

** Universidad Industrial de Santander. Facultad de Ingenierías físico-mecánicas. Escuela de Ingeniería de Sistemas e Informática.

SUMMARY

TITLE*

Instructional Design Based On Competence and Construction of a Learning Object Related to Subject Syllabus: Oriented Programming to Objects of the Systems and Computing Science Engineering.

AUTHORS**

LUZ NELLY MARTÍNEZ AYALA, HELMER LEONEL CALDERÓN ROJAS.

KEY WORDS:

Functional analysis, instructional design, curricular planning, learning object, learning styles, oriented programming to objects.

ABSTRACT:

Currently, education is based on continuous evaluation processes which offer methodological strategies focused on the use of valid and varied tools; for that reason, education is oriented to the student's developing of competences; so that, changes and revisions are made to the instructional designs, with the purpose of detecting and developing the student's skills, abilities, knowledge and competences that allow him/her to adequate to the current changes of society.

This document describes the way in which the Instructional Design was developed based on the competences of the subject Oriented Programming to Objects, for a later application to the information and communication technologies (ICT's). The developing of this methodological proposal which is oriented to the based formation on elaborated competences through a five-stage planning: Thematic contents analysis, general planning of background, setting of the relationship between purposes and contents, curricular reforming to which the identification of the formation activities, learning units reforming and formation modules identification belong to. Finally, a curricular planning is done, shaped by the criteria, contents, techniques and learning strategies (based on Felder's and Silverman's learning styles), learning evidences, techniques and evaluation instruments, duration, resources and sceneries, besides the elaboration of a learning object that supports the teaching-learning process related to the thematic "Concept of Classes and Objects".

* Degree Project

** Faculty of physical- mechanics engineering. Engineering of systems and Computing Science Engineering.

INTRODUCCIÓN

El avance conceptual y práctico de la educación ha sido posible gracias a los crecientes adelantos científicos, tecnológicos y culturales, apoyados en la filosofía del aprendizaje permanente, el reconocimiento de competencias y otros más que, en suma, se refieren al desarrollo del talento humano, producto de su inteligencia y creatividad. Todos los seres humanos gozan de un potencial para aprender, dicho aprendizaje puede ser más rápido o más lento, todo depende de la persona. El aprendizaje es más efectivo cuando las condiciones son adecuadas, existe interés en el individuo por formarse, existen estrategias de aprendizaje, estrategias de enseñanza y medios educativos que permiten organizar, guiar, supervisar, evaluar y llevar a cabo el aprendizaje.

La educación actual necesita proporcionar los medios y los recursos para dar respuesta a las nuevas formas de educación en las que están implícitas las tecnologías de información y comunicación (TIC's), el paradigma educativo y la socialización.

Para llevar a cabo este proceso es necesario el uso de diferentes recursos didácticos y electrónicos educativos, los cuales permiten guiar la enseñanza, realizar un cambio o revisión en los diseños instruccionales, con el fin de detectar y desarrollar en el alumno habilidades, destrezas, conocimientos y competencias que le permitan adecuarse al cambio actual de la sociedad; estos recursos son diseñados para apoyar el proceso de enseñanza presencial, no el aprendizaje basado en Web o Virtual, por consiguiente es importante tomar en cuenta que en la actualidad existe un proceso de aprendizaje que requiere de materiales electrónicos educativos que lo propicien; para tal fin se adaptó la metodología del análisis funcional, utilizada para la identificación de competencias y el desarrollo de Objetos de Aprendizaje.

En este documento se detalla la manera en la cual se aplicó dicha metodología para la construcción del Diseño InstruccionaI bajo la visión de competencias, de la asignatura *Programación Orientada a Objetos* correspondiente al programa académico de la Escuela de Ingeniería de Sistemas e Informática, de la Universidad Industrial de Santander. Asimismo se muestra el proceso de diseño y construcción de un objeto de aprendizaje que permita dar soporte al proceso de enseñanza - aprendizaje relacionado con la temática "*Concepto de Clases y Objetos*". En la construcción de dicho objeto se tiene en cuenta el modelo de estilos de aprendizaje de Felder y Silverman (FSLSM), utilizando las herramientas y recursos que para tal fin ofrecen las Tecnologías de Información y Comunicación (TIC's).

1. ASPECTOS GENERALES

1.1 PLANTEAMIENTO DEL PROBLEMA

Con la incursión de las Tecnologías de Información y Comunicación en los procesos de formación, se ha planteado su transformación estructural hacia la modularidad, esta transformación de contenidos va dirigida hacia la construcción de conceptos de amplio espectro y el fortalecimiento de principios básicos y finalmente, la transformación de la manera como se hace llegar, de manera que comprometan nuevas e innovadoras estrategias pedagógicas que fortalezcan el proceso enseñanza-aprendizaje.

La formación ha mejorado y ampliado su concepción y presencia. En sus inicios estaba dirigida a la creación de conocimientos, habilidades y destrezas para la vinculación a la vida laboral. Ahora, sufrió una transformación, que lo llevó de un concepto inicialista a un proceso que busca una metodología de formación continua, lo que ha ampliado su significado y alcance hacia aspectos como el desarrollo tecnológico y el complejo ámbito laboral.

En la actividad pedagógica, las metodologías de formación y gestión educativa, han evolucionado a punto que se están aprovechando de forma decidida las ventajas que brindan las Tecnologías de Información y Comunicación de tal manera que se pueda mejorar el papel que lleguen a desempeñar los profesores.

La formación basada en competencias, parte de reconocer los cambios que están sufriendo las diferentes formas de educar y enseñar. Se acerca más a la manera como se desempeña un estudiante. Pretende mejorar la manera como el estudiante adquiere su conocimiento, esto con el fin de que él maneje y afronte su papel en el proceso enseñanza-aprendizaje, y disminuir el riesgo de que en determinado momento sus conocimientos sean obsoletos. Implica “ir más allá”, es decir, superar la definición de tareas, procura ir hasta las funciones y roles que desempeñen educador y educando, de forma que conozcan los objetivos a alcanzar por cada uno de ellos.

Desde el contexto académico, las competencias son “complejas capacidades integradas en diversos grados que la institución debe formar en los individuos para que puedan desempeñarse como sujetos responsables en diferentes situaciones y contextos de la vida social y personal, sabiendo ver, hacer, actuar y disfrutar convenientemente evaluando alternativas, eligiendo las estrategias adecuadas y haciéndose cargo de las decisiones tomadas” Según Gómez de Erice¹, las competencias son “Capacidades complejas, que suponen el desarrollo de las reales posibilidades del sujeto de tomar decisiones frente a una

¹ Tomado de: [\[www.uncu.edu.ar/contenido/skins/unc/download/educacion.ppt\]](http://www.uncu.edu.ar/contenido/skins/unc/download/educacion.ppt)

situación en la que debe HACER integrando un SABER más un SABER HACER COMPLEJO”.

Para poder enriquecer y complementar la información que circula a través de la red, para convertirla en conocimiento, se debe replantear el rol que desempeña el maestro: de ser abastecedor de conocimiento, debe haber una transición hacia el hecho de ser un facilitador y mediador de conocimiento dentro de un contexto interdisciplinario.

Tratando de ser consecuentes de este fenómeno que se presenta en el medio, los estamentos educativos buscan la excelencia en la estrategia enseñanza-aprendizaje aplicando instrumentos complementarios y nuevos modelos pedagógicos cuyo objetivo es satisfacer las necesidades del estudiante y el entorno en el que se desenvuelve.

La Universidad Industrial de Santander con el ánimo de incorporar las Tecnologías de Información y Comunicación (TIC's), en sus actividades de enseñanza ha propiciado el comienzo de algunas experiencias en los últimos años tratando de introducir estas tecnologías en el ámbito educativo, dando inicio a una transformación en la manera de educación y formación universitaria. El seguimiento realizado a algunas de estas experiencias fructíferas, abre las posibilidades a que se logre una excelente y promisoría integración de las Tecnologías de Información y Comunicación en la docencia a nivel universitario.

Es nuestro deseo sumarnos a estas iniciativas con una propuesta instruccional para la asignatura de *Programación Orientada a Objetos* basado en un modelo de formación por competencias y apoyado en las Tecnologías de Información y Comunicación (TIC's), ampliando la gama de experiencias pedagógicas universitarias y potenciando desde el punto de vista de nuestra ingeniería el proceso constructivo del aprendizaje.

1.2 OBJETIVOS

1.2.1 Objetivo General. Realizar el diseño instruccional de la asignatura Programación Orientada a Objetos siguiendo la metodología de un modelo de formación basado en competencias mediado por Tecnologías de Información y Comunicación, que permita el aprendizaje significativo y personalizado (considerando estilos de aprendizaje); y construir un objeto de aprendizaje abierto e Inter-operable siguiendo los estándares de *e-learning*, que implementen el desarrollo del curriculum en contenidos relacionados con la temática *Conceptos de Clases y Objetos*.

1.2.2 Objetivos Específicos. Diseño del plan instruccional basado en competencias, mediado por Tecnologías de Información y Comunicación (TIC's) para la asignatura Programación Orientada a Objetos.

Diseño y desarrollo de un objeto de aprendizaje abierto e interoperable correspondiente al tema de Concepto de Clases y Objetos, siguiendo los lineamientos del estándar SCORM y tomando como base las estrategias pedagógicas planteadas en el diseño instruccional, que ayudarán en el aprendizaje profundo y personalizado del individuo, haciendo más agradable el ambiente educativo y motivándolo para seguir adelante según sus capacidades.

Implementación del objeto de aprendizaje construido, que dará soporte en la Enseñanza / aprendizaje de Programación Orientada a Objetos en la temática *Concepto de Clases y Objetos* utilizando el portal del profesor como medio de enlace a la Biblioteca Digital de Recursos Didácticos de la Universidad, para su inmediata exploración como material de soporte

Organización del portal Web del profesor responsable de la asignatura Programación Orientada a Objetos, en el cual se incluirá el material que el docente utiliza actualmente en la educación tradicional y al cual los alumnos tendrán fácil acceso.

1.3 JUSTIFICACIÓN

Actualmente Internet ofrece un potencial inmenso, permitiendo un sin número de aplicaciones. En el caso de la educación, se puede estimar la facilidad con que cuentan los docentes o instituciones educativas al momento de apoyarse con determinados cursos ofrecidos en línea, sobre determinados temas, generando materiales didácticos que se convierte en herramientas ergonómicas y amigables ideales para los estudiantes. Sin embargo la mala interpretación que se le da a estos cursos, ha hecho que sean medios estáticos de aprendizaje, lugares para almacenar contenidos, dejando de lado el verdadero propósito de contenidos adaptables y adaptativos, es decir que se deben considerar estilos de aprendizaje y estados de conocimiento del estudiante que ofrezcan contenidos personalizados adaptados a sus necesidades y que motiven el desarrollo de las actividades y la adquisición del conocimiento.

Como las Tecnologías de Información y Comunicación hoy en día, ofrecen un amplio espectro de recursos que facilitan el aprendizaje significativo y personalizado de conceptos complejos y la construcción y confrontación de conocimientos en ambientes interactivos y dinámicos altamente llamativos, este proyecto aspira a aprovechar las bondades del proyecto institucional "Soporte al Proceso Educativo UIS Mediante Tecnologías de Información y Comunicación (ProSPETIC)" para formular el desarrollo del diseño curricular que implemente un modelo de formación basado en competencias para dar soporte adaptativo a la enseñanza / aprendizaje de la asignatura Programación Orientada a Objetos del programa de Ingeniería de Sistemas e Informática.

La enseñanza de esta y otras asignaturas exige tanto de estudiantes como de profesores de la Universidad Industrial de Santander, disponibilidad de tiempo, espacio, fácil desplazamiento y acceso a información y material de trabajo

siempre actualizado; así como compromiso con las actividades propias del proceso educativo.

Nuestra propuesta busca diseñar, construir e implantar un objeto de aprendizaje adaptativo para la temática Concepto de clases y objetos comprendida dentro de la asignatura Programación Orientada a Objetos utilizando el portal del profesor como medio de enlace para que los alumnos tenga disponibilidad de conocer y manejar dicho objeto y como vinculo docente-estudiante que permita apoyar el proceso enseñanza-aprendizaje de la asignatura en general.

Esta propuesta está en concordancia con las pautas establecidas en el contexto general de la educación colombiana orientado a mejorar la calidad, cobertura y eficiencia del sector. Adicionalmente coincide plenamente con el proyecto educativo de la Universidad Industrial de Santander, que en su modelo institucional – Acuerdo No. 015 del 2000 – ha emprendido la transformación de sus políticas, estableciendo dentro del conjunto de estrategias: “la reforma de sus programas académicos de tal forma que los planes de las asignaturas constituyan un currículo de formación integral, y el desarrollo de nuevas metodologías pedagógicas, que vayan en pro de sus principios orientadores como lo son la formación integral y la vigencia social de los *saberes, actitudes y prácticas* construidas en el estudiantado”.

La formación profesional es un proceso que compromete a toda la sociedad, es nuestro deber como futuros profesionales iniciar proyectos o sumarnos a los existentes de forma que podamos retribuir a la comunidad universitaria y a la sociedad en general el compromiso que inicialmente ellos tomaron con nosotros, con propuestas innovadoras, económicas y viables que abonen al continuo desarrollo social y al mejoramiento académico de la Universidad.

2. MARCO TEÓRICO

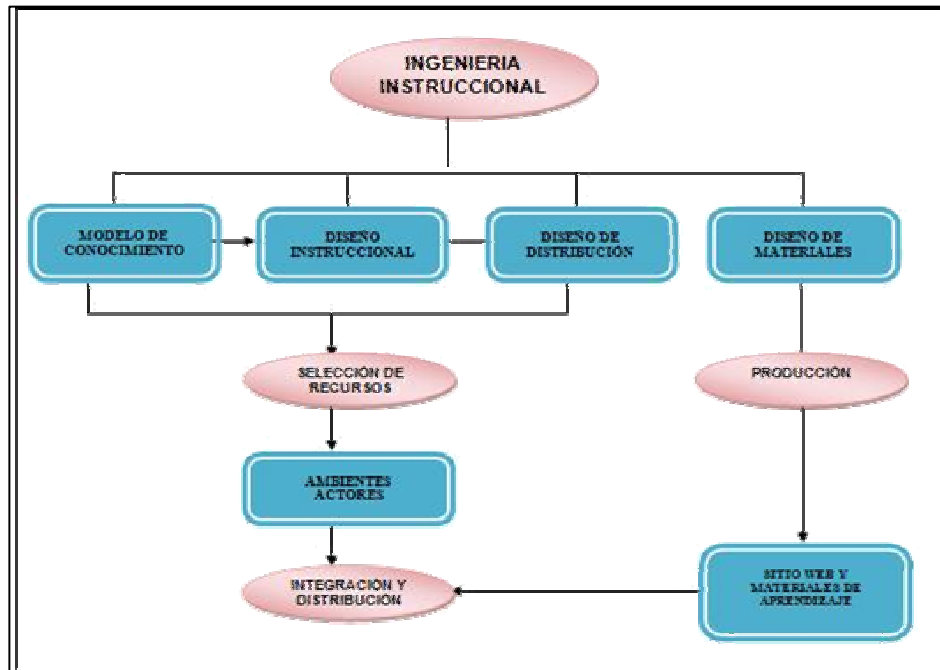


Figura 1. Ingeniería Instrucciona²

La Ingeniería Instrucciona se describe como un “método que apoya el análisis, el diseño y la entrega de la planificación de un sistema de aprendizaje, integrando los conceptos, procesos y principios del diseño instruccional, la ingeniería del software y la ingeniería de conocimiento³”

Este método en su planificación está conformado por el modelo de conocimiento, diseño instruccional, diseño de distribución y diseño de materiales.

La ingeniería instruccional aprovecha los aportes de la ingeniería del conocimiento para modelar el proceso de conocimiento; el diseño instruccional para manifestar la información adecuada de una asignatura con base en objetivos de aprendizaje y la ingeniería de sistemas de información para ofrecer el soporte tecnológico en la implementación de estrategias para el aprendizaje adaptativo y personalizado con base en estándares de e-learning.

² Adaptación hecha por el laboratorio I+D CENTIC UIS de las apreciaciones Gilbert Paquette.

³ PAQUETTE, GILBERT. Educational Modeling Languages, From an Instructional Engineering perspective.

[En línea]. Disponible en: [www.liceftel.uqebec.ca/gp/docs/Article%20EML-MISAedited.doc]

2.1 DISEÑO INSTRUCCIONAL

La propuesta metodológica de Diseño Instruccional desarrollada surge como respuesta a la inquietud de poder identificar las competencias en el contexto educativo y por ello su elaboración se orienta en los programas de formación profesionales de la universidad. El fundamento básico es el método del análisis funcional; es por eso que esta propuesta se puede precisar como una adaptación de los principios y características del análisis funcional para el contexto educativo.

En este sentido, y como la orientación del proyecto es hacia el aprendizaje significativo y personalizado se tendrá en cuenta el modelo de estilos de aprendizaje FLSM⁴ para el planteamiento de las nuevas estrategias de enseñanza/aprendizaje.

Según Peña⁵ el soporte a la enseñanza y aprendizaje de calidad ha sido uno de los aspectos críticos a tener en cuenta en escenarios de educación respaldada por las TIC's. En tales escenarios de aprendizaje y educación propuestos como respaldo para tal fin, interesa la sensibilidad que pueda tener el estudiante (esta se representa de una u otra forma en su estilo de aprendizaje) frente a los materiales educativos promovidos o creados por sus autores. Según Felder⁶ se debe ser consciente de las diferencias que tienen los estudiantes para procesar la información, con el fin de poder ofrecer materiales pedagógicos dinámicos adaptados a preferencias particulares de aprendizaje.

2.1.1 Formación basada en competencias. Surge como necesidad de adecuar la formación a las demandas de trabajo, otorga importancia a la necesidad de aprender y no simplemente asimilar conocimientos.

La formación basada en competencia es un proceso de enseñanza/aprendizaje que facilita la transferencia de conocimientos y la creación de habilidades y destrezas, desarrollando además, capacidades para aplicarlos en situaciones reales de trabajo habilitando al profesional para aplicar las competencias en diferentes contextos y en la solución de situaciones emergentes. Permite una enseñanza integral, donde se mezclan conocimientos y experiencias

Competencia: La competencia es un saber hacer con conciencia. El concepto implica que los elementos del conocimiento tienen sentido sólo en función del conjunto, en consecuencia, aunque se pueden dividir sus componentes, estos por separado no constituyen la competencia: ser competente implica el dominio de la totalidad de elementos y no sólo de alguna(s) de las partes.

⁴ Siglas en ingles del Modelo de Estilos de Aprendizaje de Felder y Silverman.

⁵ Peña, C.I., Marzo, J. L., De la Rosa, J. LI, Fabregat, R. Un sistema de tutoría inteligente adaptativo considerando estilos de aprendizaje, IV congreso iberoamericano de informática educativa, IE2002, Vigo (España), Noviembre 20-22, 2002, ISBN 848158-227-1.

⁶ M.R. Felder, Matters of Style. In ASEE Prism, 6(4), 1996, pp. 18-23.

En otras palabras, se entiende por Competencia, una construcción social compuesta de aprendizajes significativos en donde se combinan atributos tales como conocimientos, actitudes, valores y habilidades.

2.1.2 Teorías educativas. Las teorías de aprendizaje ayudan a los psicólogos a comprender, predecir y controlar el comportamiento humano, estas teorías son utilizadas para diseñar sistemas de aprendizaje en diferentes asignaturas. El propósito de las teorías educativas es el de comprender e identificar estos procesos y a partir de ellos, tratar de describir métodos para que la instrucción sea más efectiva. Es en este último aspecto en el que principalmente se basa el diseño instruccional, que se fundamenta en identificar cuáles son los métodos que deben ser utilizados en el diseño del proceso de instrucción, y también en determinar en qué situaciones estos métodos deben ser usados.

Fundamentos de las Teorías de Aprendizaje. El escenario en el que se lleva a cabo el proceso educativo determina los métodos y los estímulos con los que se lleva a cabo el aprendizaje.

Conductismo: Conductismo, corriente de la psicología que defiende el empleo de procedimientos estrictamente experimentales para estudiar el comportamiento observable (la conducta), considerando el entorno como un conjunto de estímulos-respuesta.

Cognitivismo: Se basa en los procesos que tienen lugar atrás de los cambios de conducta. Estos cambios son observados para usarse como indicadores para entender lo que está pasando en la mente del que aprende.

Constructivismo: El constructivismo en realidad cubre un espectro amplio de teorías acerca de la cognición que se fundamentan en que el conocimiento existe en la mente como representación interna de una realidad externa. El aprendizaje en el constructivismo tiene una dimensión individual, ya que al residir el conocimiento en la propia mente, el aprendizaje es visto como un proceso de construcción individual de dicho conocimiento.

Se sustenta en la premisa de que cada persona construye su propia perspectiva del mundo que le rodea a través de sus propias experiencias y esquemas mentales desarrollados.

2.1.3 Estilo de aprendizaje. Es la forma en que un individuo aprende, y como las personas tienen diferentes estilos de aprendizaje, éstos se reflejan en las diferentes habilidades, intereses, debilidades y fortalezas académicas.

El estilo de aprendizaje de Felder y Silverman (FSLSM), presenta un nuevo enfoque en el estudio de los aprendizajes que complementa y enriquece la producción que se ha hecho en este sentido.

FSLSM ofrece una buena aproximación a los criterios de categorización de estudiantes que se ha buscado para la plataforma e-escen@riUIS ya que la infraestructura actual del entorno de docencia de este ambiente, ofrece herramientas flexibles para la implementación de materiales educativos en diferentes tipos de formatos y con estrategias instruccionales adaptativas a las preferencias subjetivas de los estudiantes, promoviendo con esto el mejoramiento de la calidad de la educación mediante el aprendizaje personalizado.

El estilo FSLSM clasifica a los estudiantes según su forma de aprender de acuerdo a la siguiente lista:

Sensitivos (concretos, prácticos, orientados hacia los hechos y los procedimientos) o *intuitivos*(conceptuales, innovadores, orientados hacia las teorías).

Visuales (prefieren la presentación visual del material tal como películas, cuadros, o diagramas de flujo) o *verbales* (prefieren las explicaciones escritas o habladas).

Inductivos (prefieren la información que deviene desde lo específico hacia lo general) o *deductivos* (prefieren la información que deviene desde lo general hacia lo específico).

Activos (aprenden manipulando las cosas y trabajando con otros) o *reflexivos* (aprenden pensando acerca de las cosas y trabajando solos).

Secuenciales (aprenden poco a poco en forma ordenada) o *globales* (aprenden de forma holística).

Un mismo estudiante puede presentar varias características por lo que el docente debería ser capaz de adaptar su estilo de enseñanza a los estilos de aprendizaje de tal forma de no afectar negativamente el rendimiento del mismo o su actitud frente a los contenidos [Felder, 2004].

Para detectar el estilo de aprendizaje mediante del modelo FSLSM se ha aplicado el instrumento de diagnóstico correspondiente denominado ILS (Index of Learning Styles) que se presenta en el anexo A. En las tablas 1, 2 y 3, se presenta una distribución de los elementos de enseñanza para las cinco dimensiones del modelo dicotómico de estilos de aprendizaje, con base en las experiencias de Carver⁷ y en la estructura actual que la plataforma e-escen@riUIS permite dar a los materiales didácticos.

En la tabla 4, las herramientas de navegación se pueden adaptar para estudiantes globales, secuenciales o reflexivos.

⁷ C. A. Carver, R. A. Howard, and W.D. Lane, "Addressing Different Learning Styles Through Course Hypermedia", IEEE Transactions on Education, 42(1), February 1999, pp. 33-38.

La idea principal de realizar esta clasificación de elementos es para poder presentar los contenidos y el entorno de aprendizaje que más se acerque a la primera aproximación del estilo de aprendizaje del estudiante obtenido mediante la aplicación del cuestionario ILS del modelo FLSM⁸. Posteriormente, este perfil se refina mediante la misma interacción del estudiante con los materiales didácticos ofrecidos de acuerdo a la información percibida por los agentes monitores del sistema (ver marco científico de referencia).

Tabla 1. Estrategia Instruccional. Componentes de un curso hipermedia para los objetos de aprendizaje de una unidad docente en e-escen@riUIS

	Objetivos	Casos de estudio	Lecturas	Núcleos de conocimiento	Mapas conceptuales	Síntesis
Global	✓					✓
Secuencial					✓	
Verbal	✓		✓		✓	
Visual		✓			✓	✓
Activo				✓		
Reflexivo	✓	✓	✓		✓	
Sensitivo		✓			✓	
Intuitivo	✓				✓	

Tabla 2. Materiales Instruccionales Complementarios y Elementos de Interactividad y de Evaluación.

	Ejemplos	Animaciones	Simulaciones	Gráfico interactivo	Glosarios	Ejercicios de autoevaluación	Ejercicios de respuesta abierta
Global	✓			✓	✓	✓	✓
Secuencial	✓	✓	✓	✓	✓	✓	✓
Verbal	✓				✓	✓	✓
Visual	✓	✓	✓	✓		✓	
Activo	✓		✓			✓	✓
Reflexivo	✓	✓	✓	✓	✓	✓	✓
Sensitivo			✓	✓			✓
Intuitivo	✓	✓	✓	✓	✓	✓	✓

⁸ <http://www2.ncsu.edu/unity/lockers/users/f/felder/public/ILSdir/ilsweb.html>

Tabla 3. Formato del Material

	Diapositivas		Media clips			Texto lineal
	Texto	Multimedia	Gráficos	Video digital	Audio	
Global			✓	✓		
Secuencial	✓	✓		✓	✓	✓
Verbal	✓				✓	✓
Visual		✓	✓	✓		
Activo						✓
Reflexivo		✓	✓	✓		✓
Sensitivo		✓	✓	✓	✓	✓
Intuitivo	✓	✓	✓	✓	✓	✓

Tabla 4. Herramientas de Navegación

	Puntuales			Estructurales		Para el trabajo colaborativo		
	Flechas (avanzar Y retroceder)	Impresiones	Ayuda en línea	Mapas de Visión general	Filtros	Chat	Forum	Correo electrónico
Global				✓	✓	✓	✓	✓
Secuencial	✓	✓	✓			✓	✓	✓
Verbal	✓	✓	✓	✓	✓	✓	✓	✓
Visual	✓	✓	✓	✓	✓	✓	✓	✓
Activo	✓	✓		✓	✓	✓	✓	✓
Reflexivo	✓	✓	✓	✓	✓			✓
Sensitivo	✓	✓	✓	✓	✓	✓	✓	✓
Intuitivo	✓	✓	✓	✓	✓	✓	✓	✓

2.1.4 Tecnologías de la información y comunicación (TIC'S) en la educación. Los usos y aplicaciones de las nuevas tecnologías en los diversos campos de la actividad humana se aprovechan para lograr un aprendizaje significativo.

La educación en línea es una experiencia reciente que ha evolucionado y se ha expandido considerablemente a nivel mundial, favoreciendo el proceso de aprendizaje.

Las recientes tecnologías de la información y comunicación (TIC's), sirven para apoyar en el estudiante el acceso y la organización adecuada de la información, orientando actividades y sugiriendo estrategias que permitan al estudiante recibir, organizar y trabajar suficiente información con significado de manera que pueda activar los conocimientos que ya tiene, establecer relaciones y reelaborar conceptos, resolver problemas o tomar decisiones.

La educación respaldada con TIC's posibilita el apoyo de experiencias en los cursos presenciales, así como de los cursos de los programas en línea, sin embargo, su implementación exige compromisos y cambios a nivel pedagógico y organizativo, además del aseguramiento de una infraestructura que garantice una capacitación adecuada, completa y sistemática de los actores del proceso y la disponibilidad permanente de los recursos.

El surgimiento de nuevas tecnologías de información y comunicación (TIC's), ha creado nuevas posibilidades de desarrollo tecnológico. Asimismo, su aplicación en los procesos de enseñanza-aprendizaje con un apoyo eficiente en el manejo de la información determina estrategias de instrucción diferentes a las tradicionales clases magistrales.

La tecnología multimedia es considerada como la nueva revolución informática en el proceso enseñanza-aprendizaje. Esto se debe a su facilidad para utilizar las telecomunicaciones, creando ambientes en los que se integran los distintos medios de comunicación, tales como texto, gráficos, imágenes, sonido y video, además de un aspecto fundamental como es la interacción del usuario con el sistema.

Las ventajas de la utilización de la tecnología multimedia (cursos en línea) pueden ser consideradas desde las siguientes perspectivas:

- ▶ Facilita el aprendizaje haciendo el proceso más motivante, práctico y agradable, permitiendo el uso de presentaciones multimedia así como la exploración de documentos y otros medios.
- ▶ Permite la recopilación de trabajos, que pueden ser compartidos. Igualmente, estimula el trabajo cooperativo entre profesores y alumnos que llevan a intereses y experiencias comunes.
- ▶ Acceso para todos los miembros de la sociedad a una mayor cantidad de información actualizada y precisa, disponible en cualquier parte del mundo.
- ▶ Incremento de la interacción en el proceso educativo, lo que permite al profesor controlar; evaluar y guiar la actuación del estudiante, así como al estudiante obtener una retroalimentación cada vez que lo necesite.
- ▶ Posibilidad de trabajo al ritmo individual y a un horario conveniente.
- ▶ Permite el adiestramiento del profesorado, alumnado en las tecnologías informáticas y de comunicación, con ventajas educativas y de preparación para el mundo laboral.

En consecuencia los docentes convendrán convertirse en unos auténticos estimuladores y facilitadores de la creatividad y el aprendizaje; así como en mediadores entre el conocimiento, la información y los alumnos, garantizando que el futuro profesional será pro-activo, en consecuencia será capaz de crear nuevos conocimientos en el área de su competencia y dará nuevas soluciones, más adecuadas a los problemas que se le planteen.

2.1.5 Fundamentos de E-learning. El e-learning hace referencia al uso de las tecnologías de información y comunicación que pueden abarcarse y emplearse a través de Internet, las cuales proveen diferentes herramientas para el entorno de enseñanza-aprendizaje. El e-learning o educación en línea combina hardware y software en su implementación, pero su imagen más conocida se basa en el software bajo la figura de plataformas e-learning.

Los estándares e-learning son el vehículo a través del cual será posible dotar de flexibilidad a las soluciones e-learning, tanto en contenido como en infraestructura. Ellos han abierto una puerta hacia una manera más coherente de empaquetar los recursos y contenidos, tanto para los estudiantes como para los desarrolladores.

Esta convergencia de tecnologías e-learning es muy importante para los consumidores de estas tecnologías, debido a que los productos que se adhieran a estos estándares no quedarán obsoletos a corto plazo, protegiendo así las inversiones realizadas en ellos.

Además, estándares comunes para asuntos tales como metadata de contenidos, empaquetamiento de contenidos, secuencia de contenidos, interoperabilidad de preguntas y test, perfil de alumnos, interacción en tiempo de ejecución, etc., son requisitos indispensables para el éxito de la economía del conocimiento y para el futuro del e-learning.

E-learning se trata de una combinación de recursos, interactividad, soporte y actividades de aprendizaje estructuradas. Por lo tanto, logramos definir e-learning como: “Aquella actividad que utiliza de manera integrada y pertinente computadores y redes de comunicación, en la formación de un ambiente propicio para la construcción de la experiencia de aprendizaje”⁹

Esta actividad puede desarrollarse de forma *sincrónica*: compartiendo con el profesor o el resto de compañeros el mismo tiempo; o *asincrónica*, si las intervenciones de unos y otros quedan diferidas en el tiempo.

Los distintos estándares que se desarrollan hoy en día para la industria del e-learning se pueden clasificar en los siguientes tipos:

1. Sobre el Contenido o Curso: Estructuras de los contenidos, empaquetamiento de contenidos, seguimiento de los resultados.
2. Sobre el Alumno: Almacenamiento e intercambio de información del alumno, competencias (habilidades) del alumno, privacidad y seguridad.
3. Sobre la interoperabilidad: Integración de componentes del LMS, interoperabilidad entre múltiples LMS¹⁰.

⁹ Definición basada en el trabajo desarrollado por <http://empresas.sence.cl> con un grupo de representantes de la industria chilena de e-learning.

¹⁰ Learning Management System, Software para servidores de Internet/Intranet.

Al hablar sobre un estándar e-learning, se hace referencia a un conjunto de reglas en común que especifican cómo los fabricantes pueden construir cursos on-line y las plataformas sobre las cuales son impartidos estos cursos de tal manera que puedan interactuar unas con otras.

Estas reglas además, definen un modelo de empaquetamiento estándar para los contenidos. Los contenidos pueden ser empaquetados como “objetos de aprendizaje” (*learning objects* o LO), de tal forma que permitan a los desarrolladores crear contenidos que puedan ser fácilmente reutilizados e integrados en distintos cursos.

En definitiva, los estándares permiten crear tecnologías de aprendizaje más poderosas, y “personalizar” el aprendizaje basándose en las necesidades individuales de los alumnos.

Primordialmente lo que se quiere obtener con la implementación de un estándar e-learning es:

Durabilidad: Con tecnología desarrollada mediante el estándar evitar la obsolescencia de los cursos.

Interoperabilidad: Intercambiar información a través de una amplia variedad de LMS.

Accesibilidad: Permitir un seguimiento del comportamiento de los alumnos.

Reusabilidad: Los diferentes cursos y objetos de aprendizaje pueden ser reutilizados con diferentes herramientas y en distintas plataformas.

2.1.6 El análisis Funcional. El análisis funcional es una metodología de investigación que permite reformar luego de desarrollar una serie de etapas, las competencias que debe reunir un/a trabajador/a para desempeñarse competentemente en un ámbito de trabajo determinado¹¹ En el ámbito académico se busca identificar las competencias que debe reunir un estudiante en su proceso de formación para desempeñarse con éxito en campos específicos que se relacionan con su profesión.

El análisis funcional sigue una serie de principios y fases para cumplir el objetivo deseado el cual es la creación de un Diseño Instruccional bajo la visión de competencias.

Los principios que rigen la aplicación de la metodología del análisis funcional se concentran en tres sentencias específicas:

¹¹ CATALANO, ANA M. AVOLIO DE COLS, Susana. SLADOGNA, MONICA G. Diseño curricular basado en normas de competencia laboral: conceptos y orientaciones metodológicas. 1º. ed. - Buenos Aires: Banco Interamericano de Desarrollo, 2004. 41.

Ir de lo general a lo particular: El punto de arranque es el objetivo de la asignatura (lo general) enmarcado por las actividades de aprendizaje, seleccionadas a través del análisis de los contenidos presentes en la asignatura y combinado a su vez con la experiencia y conocimientos de los expertos docentes, expertos pedagogos y expertos en la metodología de la planeación del Diseño Instruccional que acompañen el proceso.

Identificar acciones delimitadas (discretas) manteniendo la separación de los contextos específicos: la desagregación de las actividades de aprendizaje debe ser única; poseer un inicio y un fin en su descripción, definiendo un propósito y un alcance preciso; además deben estar en consonancia con el área de estudio abarcada por la asignatura y por el programa de formación general.

Mantener una relación causa-consecuencia: este principio permite que las actividades de aprendizaje obtenidas de la desagregación sean realmente la suma de partes que den como resultado el objetivo y/o propósito origen, además que tiene la utilidad de proveer la visión de correlación que debe establecerse entre las partes.

2.2 ETAPAS DE CONSTRUCCIÓN Y DESARROLLO DE LA PROPUESTA

La presente propuesta hace parte de la metodología para el desarrollo de proyectos educativos UIS para el aprendizaje en línea del proyecto “Soporte al Proceso Educativo UIS mediante tecnologías de Información y Comunicación” PROSPETIC¹² que viene adelantando la Universidad Industrial de Santander, este proyecto está conformado diferentes fases (Ver figura 2) las cuales se explicarán a continuación:

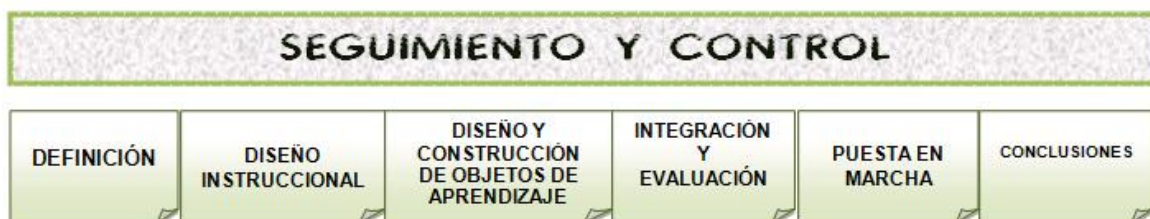


Figura 2. Fases del proyecto

2.2.1 Fase 1: Definición. Esta etapa hace alusión a la definición del proyecto, etapa en la que se identifica la necesidad actual, se justifica una solución y se planifica la labor de la propuesta.

2.2.2 Fase 2: Diseño Instruccional. Las siguientes cinco etapas determinan el

¹² Fuente y figuras tomadas de las memorias del Proyecto ProSPETIC

marco de construcción y desarrollo de la elaboración del Diseño Instruccional.

Análisis y selección de actividades de aprendizaje: Esta etapa comprende la recopilación, análisis y clasificación de los contenidos, construcción del Diagrama secuencial de Actividades de Aprendizaje.

Planteamiento de los saberes: el objetivo es realizar la desagregación de contenidos temáticos en saberes, construir la tabla de saberes, identificar las competencias a desarrollar en la asignatura.

Estructuración modular: En esta fase se enuncia e identifica las actividades de enseñanza/aprendizaje que desarrollará el estudiante, identificar las unidades de aprendizaje de la asignatura, indicar los módulos de formación de la asignatura, mantener la relación causa-consecuencia entre las diferentes agrupaciones de la estructura modular: módulos-unidades-actividades-propósitos y saberes.

Relación Actividades-Propósitos de Formación: En esta el fin es asociar cada propósito con los contenidos de la asignatura, articulando el propósito a un grupo determinado de saberes para conseguir definir las competencias conceptuales y procedimentales que debe desarrollar un estudiante con el fin de alcanzar la meta trazada por el docente para la asignatura programación orientada a objetos.

Planeación curricular: Esta etapa tiene por objetivo constituir los criterios a partir de los propósitos trazados, hacer la clasificación en contenidos temáticos, procedimentales y actitudinales, identificar las estrategias de enseñanza a utilizar según los estilos de aprendizaje propuestos.

En la tabla 5 se muestra un resumen detallado de cada una de las fases y productos del cual está constituido el diseño instruccional.

Tabla 5. Fases y productos del diseño curricular basado en competencias

ETAPAS		PRODUCTOS
Análisis de contenidos temáticos		Diagrama secuencial de Actividades de Aprendizaje
Planteamiento general de saberes y haceres.		Tabla de saberes y haceres
Estructuración modular	Unidades de aprendizaje y módulos de formación	Diagrama de módulos de formación
Asociación de Actividades y Propósitos de formación		Tabla de relación Actividades propósitos de formación
Planeación curricular		Tabla de planeación curricular

2.2.3 Fase 3: Diseño y construcción de Objetos de Aprendizaje. El objetivo de esta fase es diseñar y originar un objeto de aprendizaje identificado en la fase 2 por medio de una actividad de formación de la asignatura Programación Orientada a Objetos.

Teniendo como base una plantilla guía, un libro de estilos y la funcionalidad de los objetos de aprendizaje definidos por el proyecto ProSPETIC, posteriormente se procede a su programación de acuerdo al estándar SCORM, que permita el desarrollo de elementos interoperables y reutilizables entre plataformas de e-learning.

2.2.4 Fase 4: Integración y evaluación en la Plataforma E-Escen@Riuis. El objetivo de esta etapa es Integrar y evaluar los objetos de aprendizaje generados para la acción formativa específica. Mientras se consolida el desarrollo del ambiente en línea de aprendizaje, este proceso de integración y evaluación consistirá en la catalogación temática del producto dentro de la biblioteca digital de recursos didácticos BDRD, empleada por la universidad industrial de Santander.

La presencia del experto temático, el metodólogo del análisis funcional, los programadores del objeto de aprendizaje y el personal encargado del ambiente en línea de aprendizaje son fundamentales para el cumplimiento de este logro.

2.2.5 Fase 5: Puesta en marcha. Esta fase tiene por objeto incorporar el uso del nuevo sistema en la dinámica de trabajo y la aprobación final del proyecto: por esta razón es necesario la capacitación a los responsables del manejo garantizando la convicción y agilidad en la utilización del nuevo sistema.

Se realiza un seguimiento al aprovechamiento de las unidades docentes, mediante la interacción del estudiante con el objeto de aprendizaje para posteriormente realizar un proceso continuo de mejoramiento de herramientas con base en la información recopilada. El experto temático y los alumnos son de vital importancia para el cumplimiento de este objetivo.

2.2.6 Fase 6: Conclusiones y cierre. El objetivo de esta fase es la definición del grado de aceptación por parte de los usuarios acerca del producto que se generó, esta se lleva a cabo con la participación del estudiante o usuario final, se realizan las evaluaciones de satisfacción y control de calidad, teniendo en cuenta nuevas opciones y recomendaciones con el fin de mejorar dicho producto.

De forma reiterada, un equipo de docentes y desarrolladores del CENTIC¹³ tendrá como labor la revisión periódica para evaluar algunos de los aspectos críticos de la implantación, y hacer recomendaciones oportunas para que se aproveche cada uno de los objetos de aprendizaje.

2.2.7 Fase 7: Seguimiento y control de calidad. Se constituye un comité de seguimiento y control, que será el encargado de la supervisión y aprobación de cada una de las fases del proyecto, toma de decisiones necesarias, evaluación de resultados y aprobación. Obtener la evolución del proyecto real a nivel de tareas, compararla con la planificada, establecer la satisfacción del usuario en cuanto al servicio ofrecido y definir los puntos críticos.

2.3 HERRAMIENTAS PARA EL DESARROLLO DEL SISTEMA

2.3.1 Hardware. Las especificación mínimas de hardware para la elaboración del objeto de aprendizaje son:

- Procesador Pentium II
- Memoria RAM de 256
- Conexión de puertos USB.
- Tarjetas de sonido y video
- Unidad quemadora de CD.
- Punto de conexión a red TCP/IP

2.3.2 Software. Las herramientas de software necesarias para la elaboración del objeto de aprendizaje son:

ADOBE ACROBAT: Es un software licenciado por la Universidad que permite crear, abrir, visualizar, buscar e imprimir archivos de formato de documento portátil (PDF) con funciones de seguridad integradas.

APROA (Aprendiendo con Repositorio de Objetos de Aprendizaje): es una iniciativa en el ámbito de la Educación que propicia la adopción de tecnologías de Objetos de Aprendizaje, con el fin de crear una comunidad de desarrolladores y usuarios de objetos que por la vía de la colaboración y el intercambio de experiencias en el diseño de objetos, puedan sentar las bases de un programa de formación continua.

DREAMWEAVER: Es un editor visual de páginas dinámicas, que serán interpretadas por un servidor de aplicaciones además nos permite trabajar con paginas HTML, completadas con JavaScript y CSS, así como la inserción de contenidos cliente/servidor, con lo que podemos desarrollar

¹³ Centro de Tecnología de Información y Comunicación UIS

aplicaciones que se ejecuten en un servidor, devolviendo los datos de esa ejecución al ordenador local.

HTML : Es un lenguaje de marcación diseñado para estructurar textos y presentarlos en forma de hipertexto, que es el formato estándar de las páginas web. Gracias a Internet y a los navegadores como Internet Explorer, Opera, Firefox, Netscape o Safari.

JAVA: Es un lenguaje de programación orientado a objetos, diseñado por Sun Microsystems para el desarrollo de aplicaciones multiplataforma. Se puede describir como una versión simplificada de C++. Este lenguaje es independiente de la plataforma cliente, y las applets se ejecutan en el sistema cliente.

OFFICE: (Microsoft Office). Suite ofimática desarrollada por Microsoft, contiene programas como Word (procesador de texto), Excel (hoja de cálculos), PowerPoint (creación de presentaciones), Access (creación y mantenimiento de bases de datos), Outlook (cliente de correo electrónico), FrontPage (creación visual de páginas web), Photo Manager (editor fotográfico), Publisher (para creación de tarjetas, pancartas, etc.), etc.

VISIO: Es un programa de computadora de la empresa Microsoft, el cual sirve para realizar dibujos y diagramas diversos, siendo algunos de ellos: diagramas de oficinas, diagramas de bases de datos, diagramas de flujo de programas, UML, etc.

3. APLICACIÓN DE LA PROPUESTA METODOLÓGICA AL DISEÑO INSTRUCCIONAL BASADO EN COMPETENCIAS DE LA ASIGNATURA PROGRAMACIÓN ORIENTADA A OBJETOS

3.1 ETAPAS DE CONSTRUCCIÓN Y DESARROLLO DEL DISEÑO INSTRUCCIONAL

Para llevar a cabo el Diseño Instruccional se contó con la “Guía Metodológica: Metodología para la construcción de Diseños Instruccionales para asignaturas bajo los parámetros de ProSPETIC_{UIS}”.

Para la aplicación de esta metodología, existen cinco etapas de Diseño Instruccional que se presentan a continuación (ver figura 2):

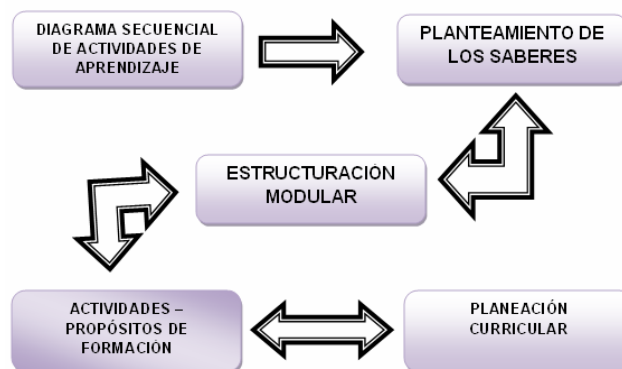


Figura 3. Fases de la construcción del Diseño Instruccional

3.1.1 Diagrama Secuencial de Actividades de Aprendizaje. El Diagrama Secuencial de Actividades de Aprendizaje tiene los siguientes objetivos:

- Representar gráficamente el entorno de la asignatura Programación Orientada a Objetos.
- Mostrar las temáticas generales identificadas y seleccionadas para la asignatura Programación Orientada a Objetos.
- Identificar relaciones entre las actividades de aprendizaje: dependencia, transversalidad, causa/consecuencia, paralelismo y preconcepto

Para la construcción del DSA² de la asignatura Programación Orientada a Objetos, es necesario estipular los siguientes lineamientos:

- Detallar los contenidos temáticos generales. (Anexo B).

- Plantear el Objetivo de Aprendizaje de la asignatura Programación Orientada a Objetos.
- Identificar las actividades de aprendizaje que modelen los alcances y lineamientos de la misma en términos del conocimiento.

► **Selección de contenidos de la asignatura.**

En el año 2006 se realizó en la Escuela de Ingeniería de Sistemas e Informática reforma a los contenidos de la asignatura Programación Orientada a Objetos, de la cual hizo parte el experto temático Jorge Herrera Castillo, por lo tanto, para la realización del DSA² nos basaremos en estos contenidos ya planteados y aprobados por la Universidad Industrial de Santander y la Escuela de Ingeniería de Sistemas e Informática.

Teniendo en cuenta el plan de estudios de la escuela de Ingeniería de Sistemas e informática (ver figura 5), se identifica como único requisito establecido para poder abordar el contenido de la asignatura Programación Orientada a Objetos la asignatura Fundamentos de Programación, la cual aborda únicamente la parte lógica y algorítmica de la programación, es por esto, que la Programación Orientada a Objetos resulta ser una base sólida para continuar con la formación académica y lograr el perfil profesional.

ING. DE SISTEMAS REFORMA

NIVEL 1			
CÓDIGO	MATERIA	CREDITOS	REQUISITOS
21274	Cultura Física y Deportiva	1	---
21211	Taller de Lenguaje	3	---
22979	Algebra Lineal	4	---
22949	Química Básica	4	---
20252	Cálculo I	4	---
22948	Fundamentos de Programación	4	---

NIVEL 2			
CÓDIGO	MATERIA	CREDITOS	REQUISITOS
22950	Física I	4	---
20253	Cálculo II	4	R:20252
22952	Biología para Ingenieros	2	R:22949
22951	Programación Orientada a Objetos	4	R:22948
22109	Etica Ciudadana	3	---
21215	Ingles I	4	---

Figura 4. Programa Académico niveles 1 y 2 de la escuela de Ingeniería de sistemas e informática¹⁴

La asignatura Programación Orientada a Objetos del programa académico

¹⁴ <http://cormoran.uis.edu.co/eisi/eisi.jsp>

de Ingeniería de Sistemas establece el siguiente contenido temas principales:

1. GENERALIDADES
2. PILARES DE LA PROGRAMACIÓN ORIENTADA A OBJETOS
3. OBJETOS Y CLASES: DISEÑO Y PROGRAMACIÓN
4. LA HERENCIA: DISEÑO Y PROGRAMACIÓN
5. EL POLIMORFISMO: DISEÑO Y PROGRAMACIÓN
6. LA PERSISTENCIA DE LOS OBJETOS
7. LAS RELACIONES ENTRE OBJETOS

Después de tener los contenidos temáticos generales de la materia, se procede a plantear el objetivo de aprendizaje de la asignatura (ver figura 4), (¿Qué debo aprender?), a lo que se quiere llegar en conocimiento, constituyendo este objetivo como la actividad de aprendizaje principal.

Objetivo de la asignatura

Entender la metodología de la programación bajo el paradigma orientado a objetos para el desarrollo de aplicaciones básicas que permitan obtener productos de calidad haciendo más eficientes los procedimientos que se están desarrollando a través de un lenguaje de programación.

Figura 5. Objetivo de la asignatura

Este objetivo debe cumplir con la estructura gramatical Verbo + Objeto + Condición.

Dentro del DSA² las actividades de aprendizaje describe de izquierda a derecha el cómo? se logra el aprendizaje y de derecha a izquierda responde al para qué? del aprendizaje alcanzado (ver figura 8).



Figura 6. Esquema general del Diagrama Secuencial de Actividades de Aprendizaje

Las actividades de aprendizaje constituyen los alcances de la asignatura en términos de conocimiento, están elaboradas de acuerdo a las siguientes

convenciones:

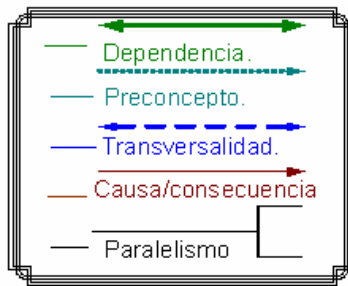


Figura 7. Convenciones empleadas en el diagrama secuencial de actividades de aprendizaje

Preconcepto: Es necesario tener conceptos de algunas asignaturas vistas con anterioridad, para poder abordar algunos temas pertenecientes a una nueva materia.



Figura 8. Preconceptos en el diagrama secuencial de actividades de aprendizaje

En el caso de programación orientada a objetos, es necesario tener estos conceptos claros provenientes de la asignatura fundamentos de programación.

Dependencia: Existen conceptos dentro de las asignaturas que se necesitan abordar mutuamente ya que se complementan entre sí, para ello se define esta relación, ya que si solo se ve uno de los conceptos el aprendizaje queda incompleto.

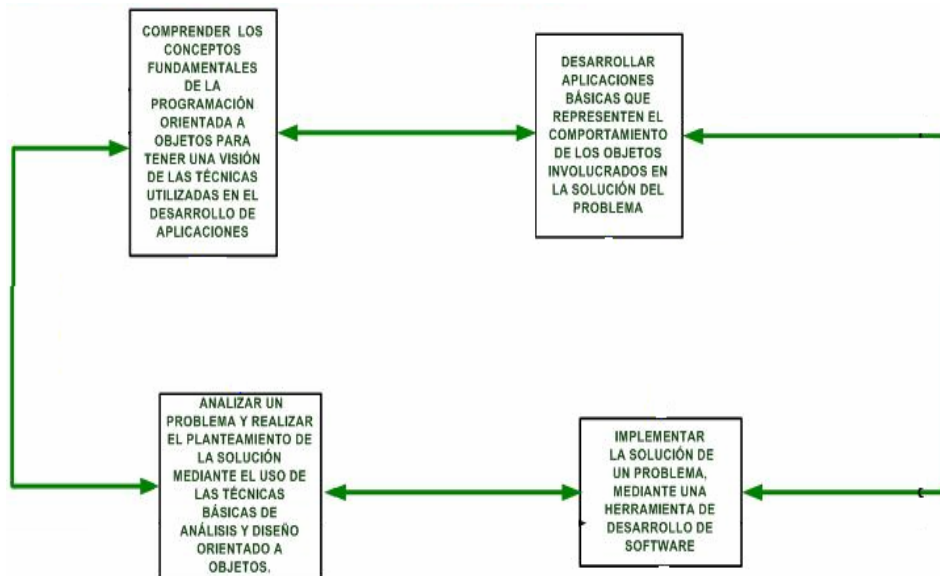


Figura 9. Dependencia en el diagrama secuencial de actividades de aprendizaje

El desarrollo de la asignatura programación orientada a objetos presenta dependencia entre las cuatro actividades de aprendizaje que conforman el DSA², es decir, comprender los conceptos fundamentales de la programación orientada a objetos es necesario para poder realizar planteamiento de soluciones mediante diagramas UML, de igual manera para implementar estas soluciones mediante el desarrollo de software se necesita tener el conocimiento de las dos actividades anteriores y para lograr desarrollar aplicaciones básicas que representen el comportamiento de los objeto es necesario abordar las tres actividades anteriores, por lo tanto, se necesita abordar cada una de estas actividades para completar el proceso de aprendizaje, si se llegara a omitir alguna de ellas, el proceso queda incompleto y por lo tanto, no se cumple con el objetivo de la asignatura.

Transversalidad: Algunos conceptos son referencia y complemento de dos o más al tiempo, aunque en contextos diferentes, por lo cual no es sencillo establecer una secuencia cronológica entre estos; por lo tanto, se puede abordar dicho contenido en varias ocasiones.

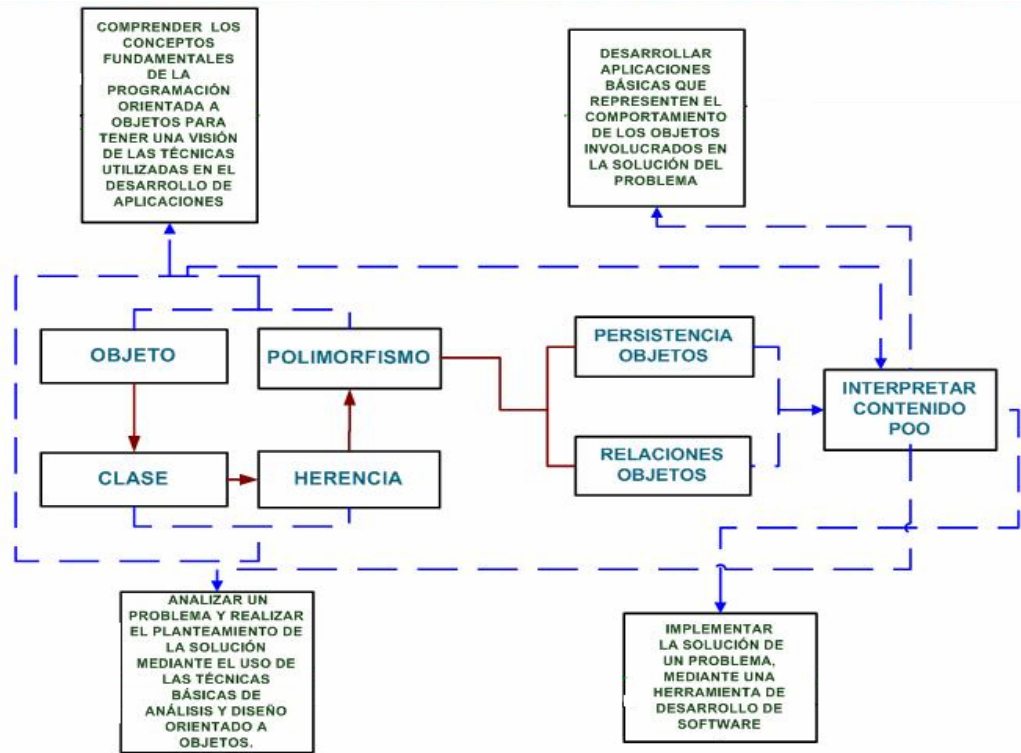


Figura 10. Transversalidad en el diagrama secuencial de actividades de aprendizaje

La asignatura programación orientada a objetos contiene siete conceptos trascendentales en su desarrollo (objetos, clases, polimorfismo, herencia, persistencia de objetos, relación entre objetos e interpretar contenido de POO), por lo tanto, son abordados durante el desarrollo de cada una de las actividades de aprendizaje, haciendo de ellos el eje central de esta asignatura, por lo tanto, es necesario que el estudiante tenga claro cada uno de estos conceptos, de tal manera se logra el proceso de aprendizaje.

Causa-consecuencia: Existen conocimientos que son causa de otros, por lo tanto, debe abarcarse primero el conocimiento origen y luego el conocimiento de destino.

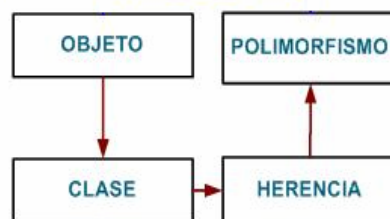


Figura 11. Causa-consecuencia en el diagrama de actividades de aprendizaje

Una vez analizado el concepto de objeto, se procede a abordar el concepto de clase, seguidamente el concepto de herencia y por último el concepto de polimorfismo. Es importante llevar la secuencia de las flechas, primero se analiza el concepto origen y en seguida se aborda el concepto destino, no es conveniente desarrollar en otro orden el análisis de estos conceptos puesto que el docente ha establecido prioridades en los conocimientos necesarios en el proceso de aprendizaje.

Paralelismo: Algunos conceptos poseen el mismo grado de importancia, por lo tanto, pueden ser vistos en cualquier orden sin afectar el proceso de enseñanza/aprendizaje.

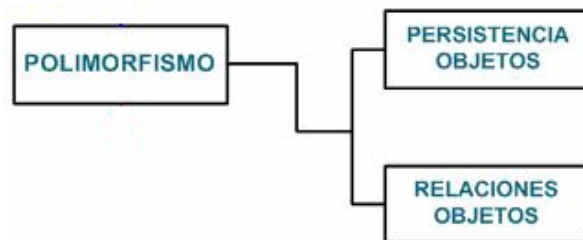


Figura 12. Paralelismo en el diagrama secuencial de actividades de aprendizaje

El concepto de polimorfismo dentro de la programación orientada a objetos incluye además el análisis de características como la persistencia y las relaciones entre objetos, cada uno de estos conceptos son necesarios abordarlos, pero el orden es indiferente en este caso, es decir, el docente puede abordar primero cualquiera de los dos conceptos y en nada se afecta el proceso de aprendizaje de la asignatura.

3.1.2 Planteamiento de los Saberes y Haceres relacionados. La siguiente fase del diseño instruccional consiste en la elaboración de la tabla de saberes, en la cual se referencian los contenidos de la asignatura programación orientada a objetos y las actividades descritas en el *DSA*²

Los saberes son acciones precisas de aprendizaje que se desarrollan de manera individual en los estudiantes, se dividen en:

Saber: Hace referencia a teorías, eventos y principios del conocimiento.

Saber Hacer: Se refiere a procedimientos, técnicas, métodos, habilidades y destrezas los cuales son necesarios desarrollar en el estudiante.

Mediante la realización de los saberes se busca:

- ▶ Clasificar los saberes en *saber* y *saber hacer*.

- Identificar las competencias individuales a desarrollar en la asignatura Programación Orientada a Objetos.

La redacción de cada saber contiene un verbo enlazado por un objeto y posteriormente su respectiva condición (ver figura 13). Los verbos empleados en esta etapa, obedecen a la clasificación realizada por Benjamín Bloom¹⁵ (Anexo A).



Figura 13. Estructura gramatical de los Saberes

La tabla de saberes presenta las siguientes características:

- ✓ La tabla muestra en forma ordenada la clasificación de los *saberes*.
- ✓ Los *saberes* describen las acciones específicas del proceso de enseñanza- aprendizaje que se desarrollará en el estudiante, y son la guía para el docente en cuanto a las directrices a desarrollar en los aprendices.
- ✓ Los *saberes* se relacionan verticalmente de forma secuencial, y en algunos casos de manera jerárquica, manteniendo siempre la relación causa-consecuencia de forma horizontal.

CONTENIDO	SABER	HACER
Tema 1	1. Saber	a. Hacer(1)
Tema 2	2. Saber	b. Hacer(1,2)
Tema 2.1	3. Saber	c. Hacer(3,9)
Tema 2.2		d. Hacer (3)
Tema 3		
Tema 4	4. Saber	e. Hacer (4)
Tema 4.1	5. Saber	f. Hacer(4,1)
Tema 5		g. Hacer (4,3)

Secuencialidad

Causa - consecuencia

Figura 14. Estructura gramatical de los contenidos desagregados

La causa-consecuencia se ve reflejada en la tabla de saberes mediante la asignación al final de cada saber hacer (entre paréntesis), el número del saber al cual pertenece, dando certeza de las relaciones utilizadas en el DSA² (transversalidad, dependencia, etc.), en la figura 15 se muestra una parte de la

¹⁵ BLOOM, Benjamín. Taxonomía de los Objetivos de la Educación: Clasificación de la Metas Educativas. Manuales I y II. 7 Ed. Buenos Aires: El Ateneo, 1979.

versión final de la tabla de saberes desarrollada para la asignatura Programación Orientada a Objetos.



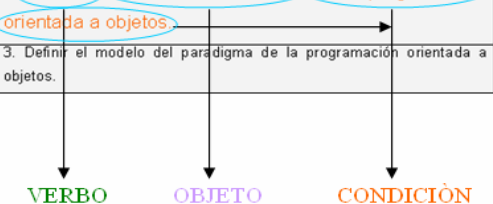
 ESCUELA DE INGENIERÍA DE SISTEMAS E INFORMÁTICA <small>ESPESHO DE INGENIERÍA DE SISTEMAS E INFORMÁTICA</small>			<small>Facultad de Ingenierías Físico Mecánicas</small>	<small>Universidad Industrial de Santander</small>	
Tabla de Saberes Asignatura: Programación Orientada a Objetos		Versión : FINAL		Página 1 de 4	
COMPRENDER LOS CONCEPTOS FUNDAMENTALES DE LA PROGRAMACIÓN ORIENTADA A OBJETOS PARA TENER UNA VISIÓN DE LAS TÉCNICAS UTILIZADAS EN EL DESARROLLO DE APLICACIONES					
SABER			HACER		
1. Recapitular conceptos de fundamentos de programación: sistemas computacionales, complejidad, abstracción, encapsulado, reutilización, calidad software, modulado, lenguajes de diseño y entornos de programación.			a. Asociar la información suministrada al entorno de programación orientada a objetos. [1]		
2. Analizar la evolución histórica de la programación orientada a objetos.			b. Comparar el manejo de los datos a través de la evolución de la programación orientada a objetos.[2]		
3. Definir el modelo del paradigma de la programación orientada a objetos.			c. Identificar las características del paradigma de la programación orientada a objetos.[3]		
					
VERBO			OBJETO		
CONDICIÓN					

Figura 15. Tabla de saberes

En el anexo D se puede consultar la tabla de saberes completa.

3.1.3 Estructuración Modular. Continuando con el desarrollo del diseño instruccional, la siguiente fase es la elaboración de la estructuración modular, en donde se enuncian e identifican las actividades de enseñanza/aprendizaje que desarrollará el estudiante, se basa en los siguientes niveles de formación:

- Módulos de formación
- Unidades de formación
- Actividades de formación
- Propósitos

► Módulos de formación, Son elementos que describen los conceptos, procedimientos, capacidades y habilidades que deben desarrollarse en torno a una temática, poseen la característica de flexibilidad para ser trasladados a diferentes contextos o asignaturas.

La figura 15, muestra uno de los módulos de la asignatura programación orientada a objetos, estos módulos se estructuran teniendo en cuenta los principios metodológicos deductivos: tener una secuencia lógica, identificar acciones delimitadas y mantener la relación de causa – consecuencia.



Figura 16. Módulos de formación de los conocimientos de la programación

► Unidades de formación, Son el referente pedagógico que permiten la organización del trabajo del docente para la orientación del proceso de aprendizaje, ya sea en aulas, talleres, laboratorios u otros entornos de formación.

Estas unidades tienen un carácter flexible ya que el docente puede fácilmente modificarlas de acuerdo a la evolución o cambios en el contenido de la asignatura.

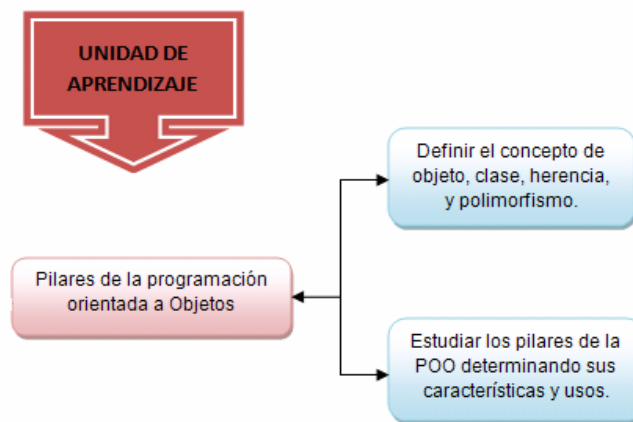


Figura 17. Unidad de Aprendizaje pilares de la programación orientada a objetos

► Actividades de formación. Estas actividades plantean los alcances que el docente define para el estudiante dentro de la unidad de formación. Estas actividades tienen un carácter flexible ya que el docente puede fácilmente modificarlas de acuerdo a las necesidades, a la evolución o a los cambios en el contenido de la asignatura.

Cada actividad presenta una estructura gramatical uniforme (verbo + objeto + condición) en la cual el verbo usado agrupa los propósitos que se desean obtener.

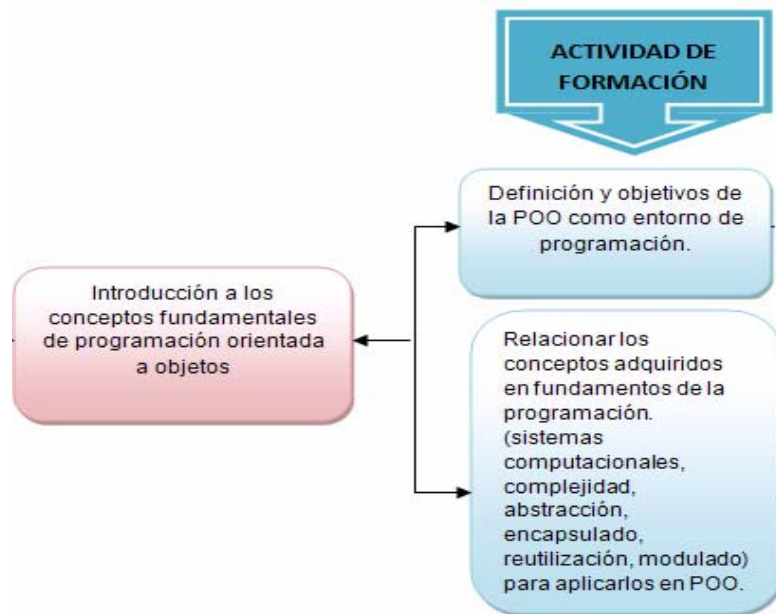


Figura 18. Actividad de formación Definición y Objetivos de la POO

► **Propósitos.** Es la última etapa de la estructuración modular (Anexo E). Los propósitos describen las finalidades o metas por las cuales se lleva a cabo una actividad de formación. Los propósitos deben contener la estructura gramatical uniforme (verbo + objeto + condición). En la figura 19 podemos observar los propósitos identificados en el primer módulo de la asignatura programación orientada a objetos.

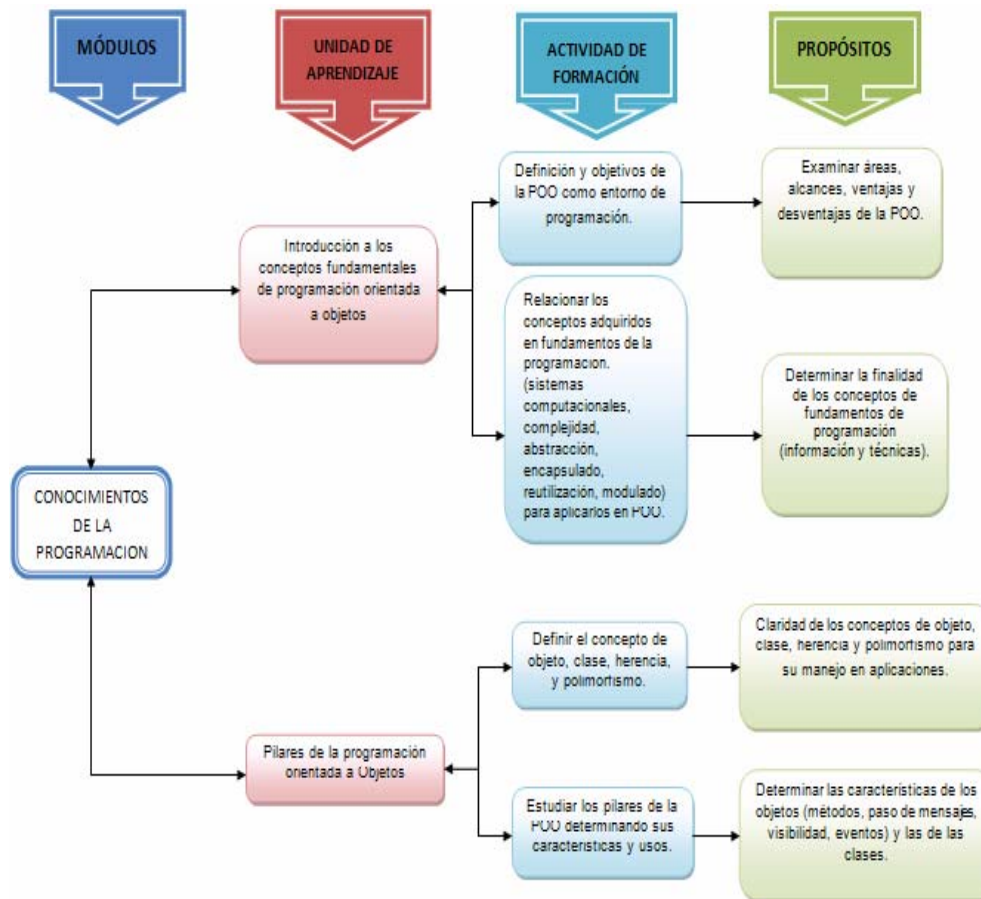


Figura 19. Niveles de la estructuración modular para el primer módulo de la asignatura POO

3.1.4 Actividades – Propósitos de formación. La obtención de esta tabla resulta de asociar cada propósito con los contenidos de la asignatura, articulando el propósito a un grupo determinado de saberes para conseguir definir las competencias conceptuales y procedimentales que debe desarrollar un estudiante con el fin de alcanzar la meta trazada por el docente para la asignatura programación orientada a objetos.





 ESCUELA DE INGENIERÍA DE SISTEMAS E INFORMÁTICA				 Facultad de Ingenierías Físico Mecánicas	 Universidad Industrial de Santander	
PROPOSITOS – ACTIVIDADES DE FORMACION PROGRAMACION ORIENTADA A OBJETOS		VERSION FINAL		NELLY MARTINEZ LEONEL CALDERON		
ACTIVIDADES	PROPOSITOS	SABERES		HACERES		
* Definir los objetivos de la P.O.O. como entorno de programación. * Relacionar los conceptos adquiridos en fundamentos de la programación. (Sistemas computacionales, complejidad, abstracción, encapsulado, reutilización, modulado) para aplicarlos en P.O.O.	* Examinar áreas, alcances, ventajas y desventajas de la P.O.O. * Determinar la finalidad de los conceptos de fundamentos de programación (información y técnicas).	1. Recapitular conceptos de fundamentos de programación: sistemas computacionales, complejidad, abstracción, encapsulado, reutilización, calidad software, modulado, lenguajes de diseño y entornos de programación. 2. Analizar la evolución histórica de la programación orientada a objetos.		a. Asociar la información suministrada al entorno de programación orientada a objetos. [1] b. Comparar el manejo de los datos a través de la evolución de la programación orientada a objetos. [2]		
* Definir el concepto de objeto, clase, herencia, y polimorfismo. * Estudiar los pilares de la P.O.O. determinando sus características y usos.	* Claridad de los conceptos de objeto, clase, herencia y polimorfismo para su manejo en aplicaciones. * Determinar las características de los objetos (métodos, paso de mensajes, visibilidad, eventos) y las de las clases.	3. Definir el modelo del paradigma de la programación orientada a objetos. 4. Establecer los pilares de la programación orientada a objetos. 5. Indicar las ventajas de la programación orientada a objetos en el desarrollo de software. 6. Describir dentro del contexto del diseño y la programación un Objeto y una Clase.		c. Identificar las características del paradigma de la programación orientada a objetos. [3] d. Analizar el funcionamiento de los pilares en el entorno de desarrollo de la programación orientada a objetos. [4] e. Describir las herramientas		

Figura 20. Tabla propósitos-actividades

3.1.5 Planeación Curricular. De esta manera se llega a la última etapa de la elaboración del diseño instruccional; la planeación curricular presenta la visión global y específica del entorno de la asignatura, provee los instrumentos para llevar a cabo sus propósitos; por lo tanto, representa un aspecto clave del diseño instruccional ya que permite construir las acciones tangibles y concretas para el desarrollo de la asignatura.

Los elementos que integran la planeación curricular se muestran en la figura 20, donde se aprecia los propósitos, los escenarios y la duración de las actividades de formación, así como las estrategias de enseñanza-aprendizaje, los métodos de enseñanza - aprendizaje, las evidencias de aprendizaje, las técnicas e instrumentos de evaluación de cada uno de los propósitos, y una guía para el desarrollo de los medios didácticos, las cuales se describirán mediante el formato de planeación curricular de la asignatura programación orientada a objetos.

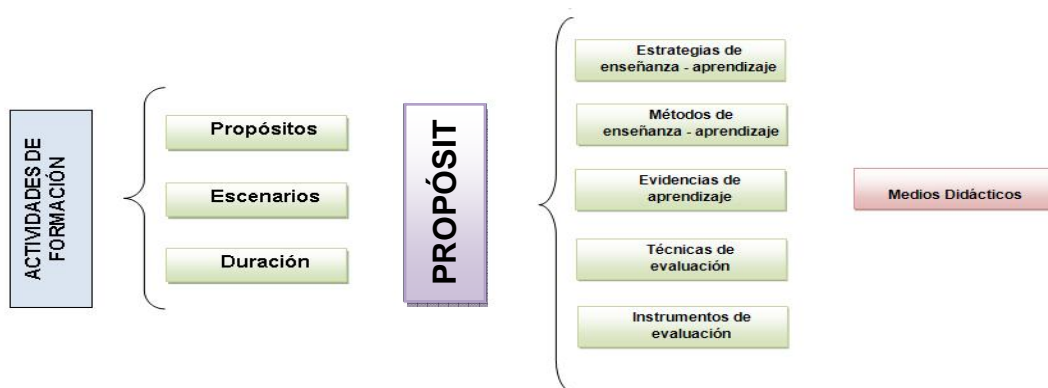


Figura 21. Elementos de la Planeación Curricular¹⁶

A continuación se analizan los elementos de la planeación curricular para el primer módulo de la asignatura programación orientada a objetos, la tabla completa de la planeación curricular de esta asignatura se puede consultar en el Anexo G.

MODULO DE FORMACIÓN	CONOCIMIENTOS DE LA PROGRAMACIÓN.
UNIDAD DE APRENDIZAJE	IDENTIFICAR LOS CONCEPTOS FUNDAMENTALES DE LA PROGRAMACIÓN ORIENTADA A OBJETOS (POO)

Figura 22. Módulo de formación de la Planeación Curricular

Escenarios, describe los lugares en los cuales se pueden desarrollar las actividades según la necesidad y requerimientos de la asignatura programación orientada a objetos (aula de clase, laboratorios, salones de conferencia, auditorios, etc.).

Duración de la actividad, hace referencia al tiempo estimado para el desarrollo de la actividad. Su valor está directamente relacionado con la complejidad de la actividad y es el experto temático quien determina el número de horas de duración.

ESCENARIOS	AULA DE CLASES, SALAS INFORMÁTICA, SALONES DE CONFERENCIA O AUDITORIOS.	DURACIÓN	12 DIAS
-------------------	---	-----------------	---------

Figura 23. Escenarios y duración del módulo de formación de la Planeación Curricular

¹⁶ Fuente: Metodología para la construcción de Diseños Instruccionales para asignaturas bajo los parámetros de PORSPETIC_{UIS}, Página 10.

Propósitos, describen las metas o finalidades que el docente define para el estudiante dentro de la unidad de formación.

Estrategias y Métodos de enseñanza-aprendizaje, donde se plantea los propósitos necesarios para cada actividad de formación y la metodología con la cual se va a desarrollar la asignatura programación orientada a objetos. Véase tabla 6.

PROPÓSITO	METODOLOGÍA DE ENSEÑANZA-APRENDIZAJE	
	ESTRATEGIA DE ENSEÑANZA APRENDIZAJE	MÉTODOS
Examinar áreas, alcances, ventajas y desventajas de la programación orientada a objetos y Determinar la finalidad de los conceptos de fundamentos de programación (información y técnicas).	1. Aprendizaje Interactivo. 2. Aprendizaje Individual. 3. Aprendizaje Colaborativo. 4. Aprendizaje por descubrimiento. 5. Aprendizaje basado en problemas.	A. Conferencia con un experto(1) B. Consulta (2,3) C. Investigaciones(3,4) D. Análisis y solución de problemas(3,5) E. Tutorial(3) F. Tareas individuales(2)

Figura 24. Propósito y metodología de enseñanza-aprendizaje del módulo de formación de la Planeación Curricular

Evidencias de aprendizaje, son las acciones que el estudiante debe estar en capacidad de realizar para demostrar lo que ha aprendido, cumpliendo el propósito planteado, alcance de la actividad de enseñanza-aprendizaje, unidad y módulo de formación, como puede observarse en la figura 21.

Las evidencias de aprendizaje son de tres tipos:

- Evidencias de conocimiento: precisan los requerimientos de conocimiento y comprensión necesarios para el cumplimiento del criterio y el aprendizaje del contenido.
- Evidencias de desempeño: hacen referencia a las técnicas y procedimientos desarrollados por el estudiante para la concreción de un aprendizaje respecto a un contenido específico. Se relacionan con la observación o demostración, intangible y tangible, del proceso de ejecución de un aprendizaje.
- Evidencias de producto: son los resultados tangibles de un proceso y proveen la evidencia de que la acción solicitada se realizó. Este tipo de evidencia mezcla los requerimientos de conocimiento y comprensión con los de técnicas y procedimientos, por lo cual sirve como evidencia de apoyo para las anteriores.

Técnicas e instrumentos de evaluación, Las técnicas e instrumentos de evaluación se relacionan mutuamente, es decir, para ciertas técnicas existen instrumentos más afines a las características de la misma, de aquí que la relación se debe hacer explícita en la planeación. La finalidad de los instrumentos se relaciona mediante anotación entre paréntesis, las técnicas de evaluación asociadas a él, estas técnicas se plantean teniendo en cuenta el propósito, el contenido temático, las estrategias y métodos de enseñanza-aprendizaje y el tipo de evidencia que se desea comprobar de la asignatura programación orientada a objetos.

EVIDENCIAS	ESTRATEGIAS DE EVALUACIÓN	
	TÉCNICAS	INSTRUMENTOS
DE CONOCIMIENTO		
Definición de la programación orientada a objetos.	1. Entrevista 2. Prueba o examen 3. Exposición	A. Cuestionario Formal(1) B. Cuestionario(1,2) C. Informe(3)
Objetivos de la programación orientada a objetos.	1. Entrevista 2. Prueba o examen 3. Exposición	A. Cuestionario Formal(1) B. Cuestionario(1,2) C. Informe(3)
Ventajas y desventajas de la programación orientada a objetos como herramienta de programación.	1. Entrevista 2. Prueba o examen 3. Exposición	A. Cuestionario Formal(1) B. Cuestionario(1,2) C. Informe(3)
Conceptos en Sistemas Computacionales, Complejidad, Abstracción, Encapsulado, Reutilización, Modulado para su manejo en aplicaciones.	1. Ensayo 2. Prueba o examen 3. Actividades complementarias 4. Práctica de laboratorio 5. Diagramas de información	A. Ensayo(1) B. Ejercicios(2,3) C. Algoritmo(4, 5) D. Taller de Problemas(2, 3)
DE DESEMPEÑO		
Bica la programación orientada a objetos entre los lenguajes más confiables y completos.	1. Entrevista 2. Prueba o examen 3. Exposición	A. Cuestionario Formal(1) B. Cuestionario(1,2) C. Informe(3)
Información y técnicas de fundamentos de programación que sirven de habilidades en el desarrollo de aplicaciones POO.	1. Prueba o examen	A. Cuestionario(1) B. Ejercicios(1)

Figura 25. Técnicas e instrumentos de evaluación del módulo de formación de la Planeación Curricular

Tabla 6. Estrategias y técnicas de enseñanza/aprendizaje

ESTRATEGIAS	TÉCNICAS	
Aprendizaje interactivo	Presentación participativa Exposición Conferencia por un experto Entrevista Panel Debate Formulación de preguntas	Phillips 6.6 Visitas Foro de discusión Mesa redonda Simposio Cine foro, foro teatro o disco foro.
Aprendizaje individual	Consulta Reporte Elaboración de ensayo Tareas individuales Resumen	Laberintos de acción Análisis e interpretación de lectura Análisis y resolución de problemas
Aprendizaje colaborativo	Consulta Resumen Análisis e interpretación de lectura Análisis y resolución de problemas Taller de ejercicios Exposición Técnica del rompecabezas	Investigación Proyecto Panel Debate Seminario Concurso Juego de roles Lluvia de ideas Tutorial
Aprendizaje por descubrimiento	Práctica de laboratorio Proyecto Investigaciones	
Aprendizaje basado en problemas	Análisis de ejercicios Resolución y análisis de ejercicios	Análisis y resolución de problemas Simulaciones
Aprendizaje significativo	Solución de casos Analogía Resumen Organizador previo Ilustraciones Mapas conceptuales	Redes semánticas Mapa mental Diagramas Lluvia de ideas Formulación de preguntas

Tabla 7. Técnicas e instrumentos de evaluación¹⁷

TÉCNICA	INSTRUMENTOS	
Observación	Lista de verificación Ficha de observación	
Entrevista	Cuestionario formal Cuestionario informal	
Debate	Anecdótico Resumen Toma de notas	
Mesa redonda	Anecdótico Toma de notas Resumen Cuestionario informal	
Exposición	Lista de verificación Informe Anecdótico Toma de notas	Resumen Relatoría Preguntas informales
Ensayo	Ensayo Lista de verificación	
Prueba o examen	Cuestionario Taller de problemas Ejercicios Test	
Mapa conceptual	Mapa conceptual	
Diagramas de información	Mapa mental Cuadro sinóptico Esquema Redes semánticas	Algoritmo Panel de información Tablas
Proyectos	Informe Productos asociados Portafolio	
Actividades complementarias	Relatorías Resumen Ejercicios	Taller de problemas Visitas técnicas
Seguimiento de actividades	Encuestas Bitácoras Registro de actividades	Anecdótico Auto evaluación Coevaluación
Práctica de laboratorio	Informe Lista de chequeo Cuestionario Algoritmo Anecdótico	

¹⁷ Esta recopilación se tomó de la tesis cuyo autor y nombre son: ESTRADA DÍAZ, Lilia Yarley. Elaboración y documentación de una propuesta de diseño curricular bajo la visión de competencias para la asignatura mediciones eléctricas y estudio de su implementación en una plataforma e-learning. Bucaramanga 2005. Trabajo de grado (Ingeniera Electrónica) Universidad Industrial de Santander. Escuela de Ingenierías Eléctrica Electrónica y de Telecomunicaciones. Capítulo IV, pág. 140.

Medios Didácticos, para dar cumplimiento con la finalidad de un objeto de aprendizaje, es viable utilizar diversos recursos digitales como textos, audio, videos, animaciones, etc., recursos que diseñados con bases metodológicas aseguran un soporte adecuado al estudiante y al docente en el proceso de enseñanza-aprendizaje.

DISEÑO DE LOS MEDIOS DIDÁCTICOS PARA EL OBJETO DE APRENDIZAJE	
NUCLEO DEL CONOCIMIENTO	Documentos como introducción la poo, el paradigma de la poo, composición de una aplicación, construcción de una aplicación.
PDF	Construcción de programas. Conceptos generales. Fundamentos de la programación
VIDEO	Alcances de la programación orientada a objetos Metas de la programación orientada a objetos.
AUDIO	Construcción de programas. Conceptos generales. Fundamentos de la programación.
GRAFICO	Diagramas flujo, diagramas que muestren el uso de los conceptos
APLICATIVO	Aplicación que interactúe con las características de los conceptos de fundamentos de la programación y que ilustre un posible resultado y su relación con la poo

Figura 26. Medios didácticos para el objeto de aprendizaje del primer módulo de la planeación curricular

4. METODOLOGÍA PARA LA GENERACIÓN Y ENCAPSULAMIENTO DEL OBJETO DE APRENDIZAJE

En este capítulo se expone la metodología utilizada para la generación y el encapsulamiento del objeto de aprendizaje; esta metodología fue propuesta por el equipo de profesionales pertenecientes al Proyecto FONDEF “Aprendiendo con Repositorios de Objetos de Aprendizaje”, APROA¹⁸.

4.1 DEFINICIÓN Y CARACTERÍSTICAS DEL OBJETO DE APRENDIZAJE

Un objeto de aprendizaje es una entidad digital basada en la aplicación de la metodología del análisis funcional para programas de formación por competencias (diseño instruccional) que puede ser utilizado, reutilizado o referenciado durante el aprendizaje en línea con el objetivo de generar conocimientos, habilidades y destrezas en función de las necesidades del estudiante..

Características de un Objeto de Aprendizaje

Un objeto de aprendizaje debe conservar las siguientes características para garantizar su eficiencia:

Ser autocontenido, es decir, por si solo debe ser capaz de dar cumplimiento al objetivo propuesto. Solamente puede incorporar vínculos hacia documentos digitales que profundizan y/o complementan algunos conceptos del contenido.

Ser interoperable, debe contar con un estándar internacional de interoperabilidad SCORM (Sharable Content Object Reference Model) para efectos del proyecto, garantizando su utilización en plataformas con distintos ambientes de programación.

Ser reutilizable, es decir, debido a que pretende dar cumplimiento a un objetivo específico, podrá ser utilizado por diversos educadores bajo distintos contextos de enseñanza.

¹⁸ <http://www.aproa.cl/1116/article-68370.html>

Manual de buenas prácticas para el desarrollo de un objeto de aprendizaje; Este manual define la metodología para la creación de objetos de aprendizaje desarrollado por un grupo de profesionales chilenos.

Ser durable y actualizable en el tiempo, es decir, debe estar respaldado por una estructura(repositorio) que permita, en todo momento, incorporar nuevos contenidos y/o modificaciones a los existentes. De esta forma un objeto debe evitar ser obsoleto con el paso del tiempo.

Ser de fácil acceso y manejo para los alumnos, es decir, la misma estructura de respaldo debe facilitar a los alumnos el acceso al objeto así como el manejo de éste en el aprendizaje.

Ser secuenciable con otros objetos, es decir, la estructura de respaldo debe posibilitar la secuenciación del objeto con otros bajo un mismo contexto de enseñanza.

Ser breve y sintetizado, es decir, debe alcanzar el objetivo propuesto mediante la utilización de los recursos (textos, imágenes, diagramas, figuras, videos, animaciones, otros) mínimos necesarios, sin extremar en la saturación de recursos y en la carencia de los mismos.

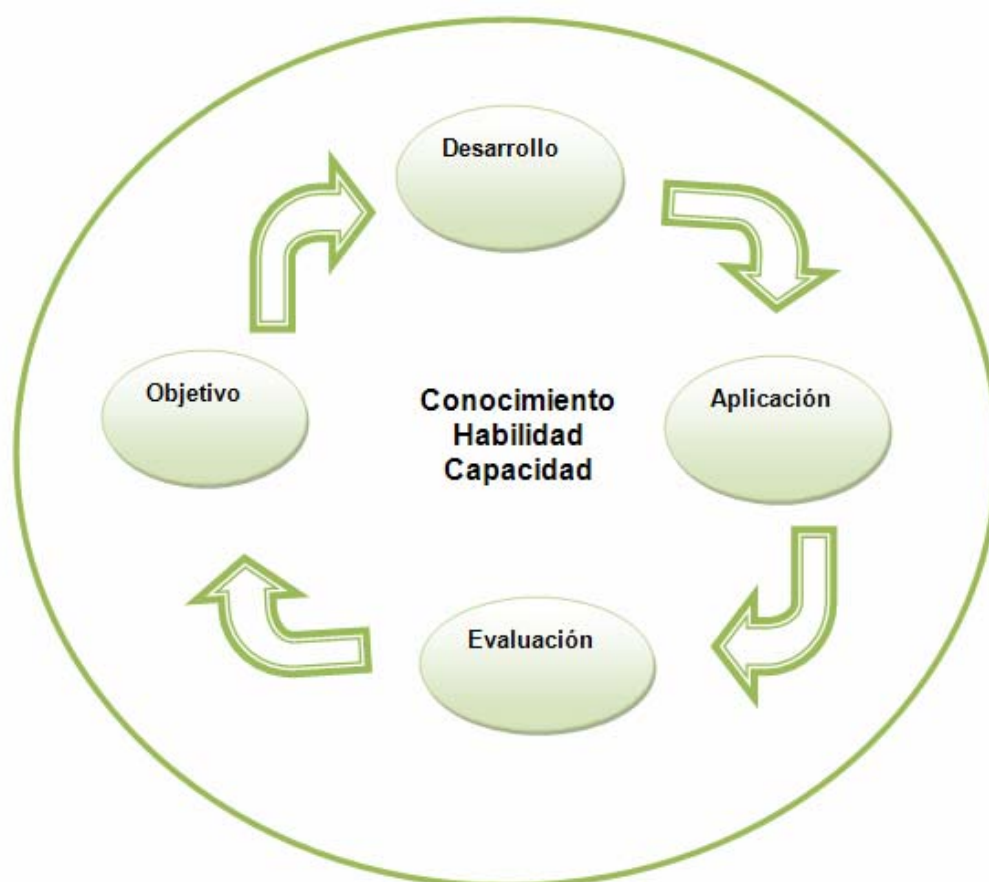


Figura 27. Estructura de un Objeto de Aprendizaje

4.1.1 Nombre del Objeto de Aprendizaje. El nombre del objeto de aprendizaje debe representar de forma clara y simple el contenido de la temática, evitando la ambigüedad en la idea.

4.1.2 Objetivo del Objeto de Aprendizaje. Dependiendo del nivel de globalidad del objetivo planteado por un objeto de aprendizaje se encuentran 3 niveles:

Objeto de aprendizaje global (OAg), aquel que presenta un objetivo general, que puede ser la base para el desarrollo de objetos con objetivos más específicos.

Objeto de aprendizaje temático (OAt), aquel que presenta un objetivo orientado a un tema específico, que puede permitir el desarrollo de objetos aún más específicos.

Objeto de aprendizaje específico (OAe), aquel que presenta un objetivo orientado a un aspecto específico de un tema, siendo el escalafón más alto en especificidad de objetivos.

4.1.3 Contenido del Objeto de Aprendizaje. Para cumplir el objetivo planteado en un objeto de aprendizaje, es viable hacer uso de diversos recursos digitales, tales como textos, imágenes, diagramas, gráficos, figuras, videos, narración, animaciones, pdf, los cuales deben ser organizados metodológicamente asegurando un óptimo aprendizaje por parte del alumno.

Para llevar a cabo el desarrollo del contenido del objeto, se hace necesaria la implementación de plantillas que permitan facilitar el diseño del mismo, economizando tiempo y recursos en la generación de objetos, y facilitando la secuenciación de estos bajo un mismo contexto de enseñanza.

4.1.4 Conexión entre el Objeto de Aprendizaje y el Diseño Instruccional. La construcción del objeto de aprendizaje presenta amplia relación con el diseño instruccional, puesto que en este, se establece lo que el estudiante debe aprender y el objeto de aprendizaje proporciona los medios para llevar a cabo este proceso.

4.1.5 Aplicación del Objeto de Aprendizaje. Debido a que un objeto de aprendizaje debe ser capaz de afianzar el proceso de enseñanza de un objetivo por sí solo, debe incorporar una aplicación ó experiencia que

permita al alumno aplicar el conocimiento aprendido, ya sea bajo ambientes reales o simulados.

4.1.6. Evaluación del Objeto de Aprendizaje. Un objeto puede incorporar diversos métodos de evaluación, tales como preguntas de asociación, completar, ordenar, selección, sopa de letras, cuestionario académico, pregunta abierta ó algún otro de ejercicio que le asegure al profesor una correcta evaluación del contenido aprendido por el alumno.

Independiente del tipo de evaluación incorporada en el objeto, cada problema planteado debe tener su retroalimentación y un resumen del rendimiento logrado.

4.1.7 Vínculos de Profundización del Contenido. Es aconsejable que todo objeto agregue vínculos ó direcciones de referencias digitales que posibiliten al alumno profundizar y/o complementar el contenido transmitido por el objeto.

4.1.8 Declaración de Autoría del Contenido. El contenido presentado por un objeto de aprendizaje debe declarar la autoría del o de los profesores que participaron en la generación del objeto, citar las fuentes de los textos, imágenes, gráficos, videos, o cualquier otro recurso incorporado que no haya sido dispuesto por el profesor.

4.2 PROCESO DE GENERACIÓN DE UN OBJETO DE APRENDIZAJE

La comunidad APROA, tiene a disposición una serie de herramientas para desarrollar objetos de aprendizaje, los cuales fueron utilizados en el avance de este proceso.

Para iniciar la generación de un objeto, es necesario definir e incorporar el objetivo directamente en la plataforma APROA. Esto se realiza diligenciando un formulario con preguntas acerca de las características básicas del objeto, tales como el título, el editor, el desarrollador de contenido, el desarrollador de multimedia, la clasificación temática del contenido, y la fecha de incorporación, entre otros, información que forma parte del catálogo de objetos presente en APROA para su clasificación y consulta.

Posteriormente, APROA propone hacer la recolección del material que irá en el objeto de aprendizaje, para esto, el profesor debe desarrollar el contenido en un editor de texto, el cual, una vez finalizado, se irá incorporando secuencialmente, de manera paralela definir los recursos multimedia (imágenes, animaciones, videos, narración, gráficos, otros) que se incorporarán al objeto.

Completo este proceso de selección de recursos, esta información debe ser incorporada a la plataforma a través de mecanismos simples de adjuntado de archivo Flash.

La aplicación y la evaluación deben desarrollarse directamente en la plataforma, para lo cual ésta incorpora herramientas especiales de edición y elaboración de métodos de evaluación.

Una vez incorporados todas las secciones del objeto en la plataforma APROA, automáticamente ésta generará el metadato del objeto y los patrones SCORM, empaquetando así el objeto definitivo.

Con base en esta serie de pasos propuestos como metodología en la comunidad APROA, se da inicio a la construcción del objeto de aprendizaje para este trabajo de grado. Para ello se utilizó la plantilla¹⁹ estándar de objetos de aprendizaje para la Plataforma e-escen@riuis

La estructura de la plantilla se basa en páginas de contenidos, las que liberan al alumno de la sobrecarga de información por pantalla de lectura. De esta manera, el alumno puede gestionar el contenido fomasimilar al formato de un libro. La plantilla dispone de botones que permiten regular el volumen y el inicio narraciones, botones que permiten avanzar o retroceder en cada una de las páginas que la componen.

EL material fue incorporado partiendo del contenido de la materia, con sus temas y subtemas, luego se adicionaron los núcleos de conocimiento, cada uno con su respectivo soporte. de igual manera se generan los documentos multimedia como lo son las animaciones y los videos, cada uno de estos tiene su correspondiente link que permite ser visualizados y tratados de forma clara. Seguidamente se plantea la evaluación de los temas mediante ejercicios y actividades de formación como foros, con el fin de valorar el aprendizaje del estudiante.

Después de tener cada uno de estos materiales completos y aprobados por el experto temático, y el metodólogo, se procede a empaquetarse en SCORM mediante la herramienta Reload siguiendo los estándares establecidos por el mismo dando una secuencia lógica al empaquetamiento para poder ser utilizado mediante cualquier tecnología.

¹⁹ Plantilla desarrollada por el Laboratorio de I+D del CENTIC.

5. DISEÑO Y DESARROLLO DEL OBJETO DE APRENDIZAJE QUE SOPORTA LA TEMÁTICA CONCEPTO DE CLASES Y OBJETOS

5.1 CONCEPTO Y CARACTERÍSTICAS DEL OBJETO DE APRENDIZAJE

De acuerdo a la metodología descrita anteriormente para la creación del objeto de aprendizaje, este capítulo describe la implementación sobre la plataforma educativa institucional e-escen@riuis, desarrollo de la universidad Industrial de Santander.

El objeto de aprendizaje consta de algunas características necesarias para garantizar su eficiencia.

- a) Ser autocontenido, muestra los contenidos necesarios para cumplir con los propósitos u objetivos relacionados con la temática “Conceptos de clases y objetos”.
- b) Ser interoperable, el objeto cuenta con la aplicación del estándar internacional de interoperabilidad SCORM, garantizando su funcionamiento en diversas plataformas de e-learning que soporten el estándar.
- c) Ser reutilizable, el objeto puede ser utilizado en diferentes contextos según los propósitos planteados en cada uno de ellos.
- d) Ser durable y actualizable en el tiempo, este objeto permite ser actualizado fácilmente, teniendo la posibilidad de agregar contenidos nuevos, complementarlos y modificar los contenidos existentes.
- e) Ser de fácil acceso y manejo, el acceso a este objeto principalmente será para los alumnos que estén cursando la asignatura, presenta una interfaz sencilla facilitando el procesos de aprendizaje.
- f) Ser secuenciable con otros objetos, el objeto fue diseñado pensando en la secuencialidad con los objetos de aprendizaje de proyectos posteriores de la asignatura programación orientada a objetos.
- g) Ser breve y sintetizado, el objeto de aprendizaje es puntual y utiliza recursos como documentos pdf, animaciones y videos para dar soporte, por lo tanto no se excede en contenido, pero tampoco carece de él, logrando un equilibrio de la información.

h) En el objeto de aprendizaje se encuentra incorporada cada una de las fuentes de donde se tomó la información que hace parte de la construcción del objeto, consolidando el cumplimiento de las leyes de derecho de autor existentes.

Para finalizar con esta primera etapa, la metodología utilizada propone inicialmente una descripción del Objeto de Aprendizaje, el cual se detalla a continuación:

5.1.1 Nombre del Objeto de Aprendizaje. El nombre del objeto de aprendizaje es “Concepto de clases y objetos”, esta temática presenta de forma clara y sencilla la representación del mismo logrando así facilitar el proceso de Enseñanza/aprendizaje.

5.1.2 Objetivos del Objeto de Aprendizaje. El objeto de aprendizaje se clasifica como un objeto de aprendizaje temático (OAt), es decir, aquel que presenta un objetivo orientado a un tema específico.

Objetivos

- ▶ Incentivar al estudiante para adquirir los conocimientos básicos de la temática planteada.
- ▶ Describir la importancia de las clases y los objetos en la programación orientada a objetos.
- ▶ Definir el diseño básico de las clases y objetos dentro de los lenguajes de programación orientados a objetos.

La temática Concepto de clases y objetos está compuesta por los siguientes subtemas:

- Introducción
- Clases y Objetos en el contexto de diseño y programación
- Los Métodos
- La Visibilidad
- El paso de Mensajes
- Eventos
- Diseño y programación de aplicaciones orientado a objetos

Para la implementación del material didáctico que forma parte del objeto de aprendizaje, fue necesaria la utilización de tecnologías multimedia para el manejo de gráficos, videos, textos, animaciones y aplicaciones.

5.1.3 Contenido del Objeto de Aprendizaje. Para llevar a cabo el desarrollo del contenido del objeto, se hace necesaria la implantación del objeto de aprendizaje en una plantilla a partir de la cual se podrá acceder a los recursos digitales, tales como pdf, imágenes, gráficos, videos, narraciones, animaciones, entre otros.

Para la implantación del objeto de aprendizaje se dispone de una plantilla en la plataforma educativa institucional *e-escen@riUIS* realizada por la Universidad Industrial de Santander, la cual favorece el trabajo de diseño del objeto y el proceso de secuenciación y comprensión del contenido por parte de los alumnos.

A continuación se hace una descripción de la plantilla y sus componentes.

Para acceder a cada uno de los contenidos del objeto de aprendizaje, se dispone de la ventana de contenidos (ver figura 28), en la cual se encuentran organizadas de manera tal que se identifique el tema general, y cada uno de los subtemas o temas específicos que lo complementan.



Figura 28. Tabla de contenido del Objeto de la asignatura POO

Una vez seleccionado el contenido a consultar, se despliega la ventana principal (ver figura 29), consta de varios botones los cuales permiten al estudiante la navegación y el uso del objeto de aprendizaje. A continuación se hace una descripción de cada uno de ellos:

Núcleo de conocimiento: Es la parte central donde se muestra una breve descripción del tema, (síntesis, explicaciones, animaciones, imágenes, etc.)



Figura 29. Núcleo de Conocimiento

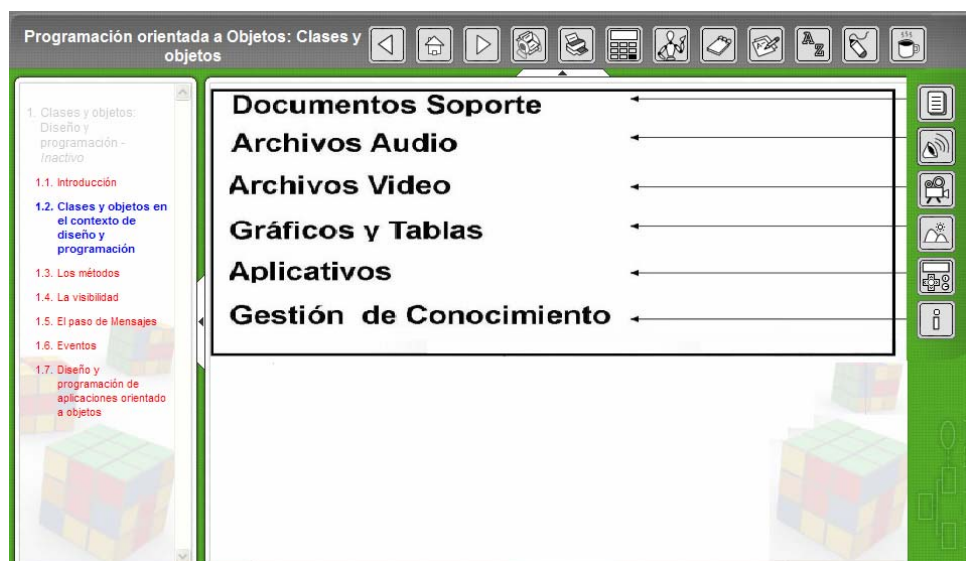


Figura 30. Explicación de la plantilla Web para el objeto de aprendizaje



Documentos de soporte: En este botón se encuentra el material que da soporte al núcleo de conocimiento. Todos los documentos se

realizaron en formato PDF.



Archivos Audio: Los archivos de audio se utilizan para expresar de forma oral y breve el contenido de la temática o subtema tratado. Los formatos de los archivos de sonidos que se utilizaron fueron: .mp3, .wav (formato comprimido) y .wma.



Archivos Videos: Estos permiten al estudiante de forma visual interpretar el contenido relacionado con la temática. Los formatos de los archivos de video fueron .avi ó .mpg (abreviatura de mpeg). Además se utilizó para la edición de los videos codecs estándar.

Animaciones: Las animaciones deben hacer alusión a alguna explicación de forma gráfica al contenido textual de la temática, las fuentes y fondos deben ser guiados de acuerdo a la hoja de estilos de e-escen@riuis, se desarrollaron flash y .avi



Gráficos y Tablas: Los gráficos deben hacer alusión a la explicación de la temática, se debe tener en cuenta las siguientes condiciones para su elaboración:

Los gráficos que se empleen se les deben hacer tratamiento para que no sean tan pesados a la hora de cargarlos en la plataforma (Calidad Vs. Tamaño).Las extensiones de las imágenes utilizada fue .sw.



Aplicativos: Aquí se encuentra el software de soporte para dar una explicación práctica que permite al usuario interactuar con este, mediante el ingreso de datos y comprobación de un proceso de la temática, para su elaboración se tuvo en cuenta las siguientes condiciones: El aplicativo debe contener una ayuda muy clara para el usuario interactúe con el de una forma fácil y cómoda. El aplicativo debe ser muy intuitivo, que el usuario no se pierda. Las fuentes y estilos deben ser guiados de acuerdo a la hoja de estilos de e-escen@riuis.

Debe ser desarrollado en aplicaciones orientadas a la web como java.



Gestión de Conocimiento: Es aconsejable que todo objeto incorpore vínculos ó direcciones de referencias digitales que permitan al alumno profundizar y/o complementar el contenido entregado por el objeto.

A continuación se hace una descripción de parte del material que da forma al objeto de aprendizaje desarrollado en el presente proyecto

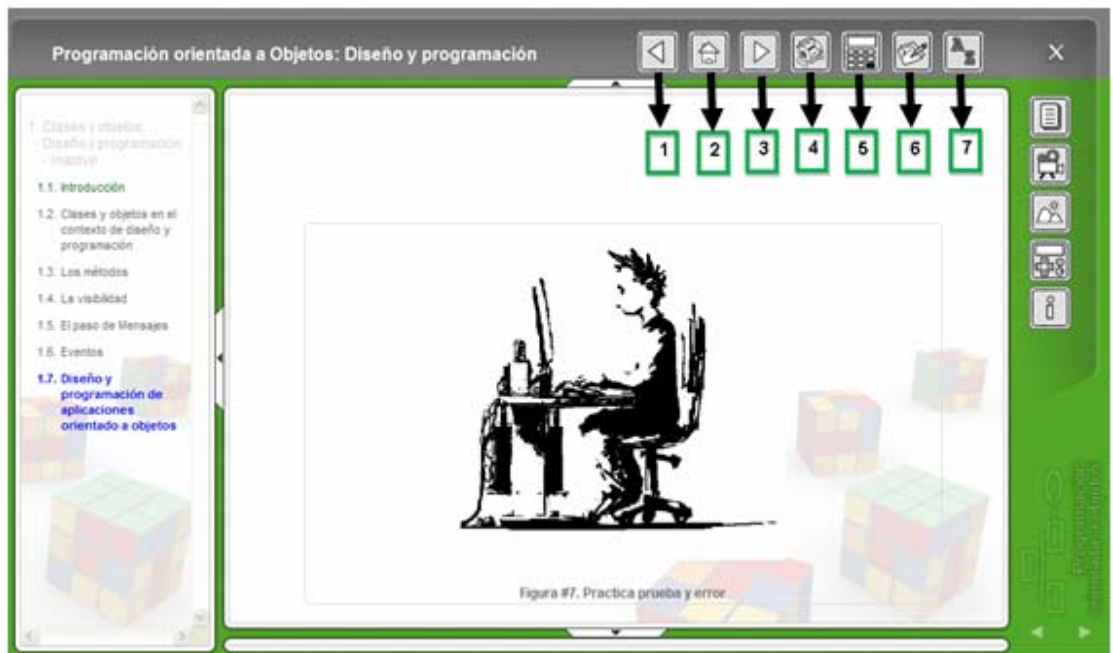


Figura 31. Explicación de los elementos adicionales para el objeto de aprendizaje

1. Atrás: Permite dirigirse al contenido anterior.
2. Home: Permite volver a la página inicial.
3. Adelante: Permite avanzar al contenido siguiente.
4. Bibliografía: Contiene los documentos necesarios para la comprensión de la temática.
5. Calculadora: Permite realizar cálculos dentro de la plantilla
6. Ejercicios: Se encuentran ejercicios de las diferentes temáticas con el fin de evaluar los conocimientos que el alumno adquirió a través del estudio de los contenidos.
7. Glosario: Esta opción permite ver el significado de las palabras desconocidas tratadas en las temáticas.

A continuación se presenta una breve descripción del contenido de la plantilla de la asignatura programación orientada a objetos.

✓ INTRODUCCIÓN

* Objetivos Específico

Reconocer los conceptos teóricos que sirven de pilares para la programación orientada a objetos.

Se plantearon los siguientes propósitos:

- *Claridad de los conceptos de objeto y clase
- * Determinar las características de los objetos (métodos, paso de mensajes, visibilidad, eventos) y las de las clases.

Se plantearon las siguientes actividades:

- * Definir los objetivos de la POO como entorno de programación.
- * Relacionar los conceptos adquiridos en fundamentos de la programación. (Sistemas computacionales, complejidad, abstracción, encapsulado, reutilización, modulado) para aplicarlos en POO.

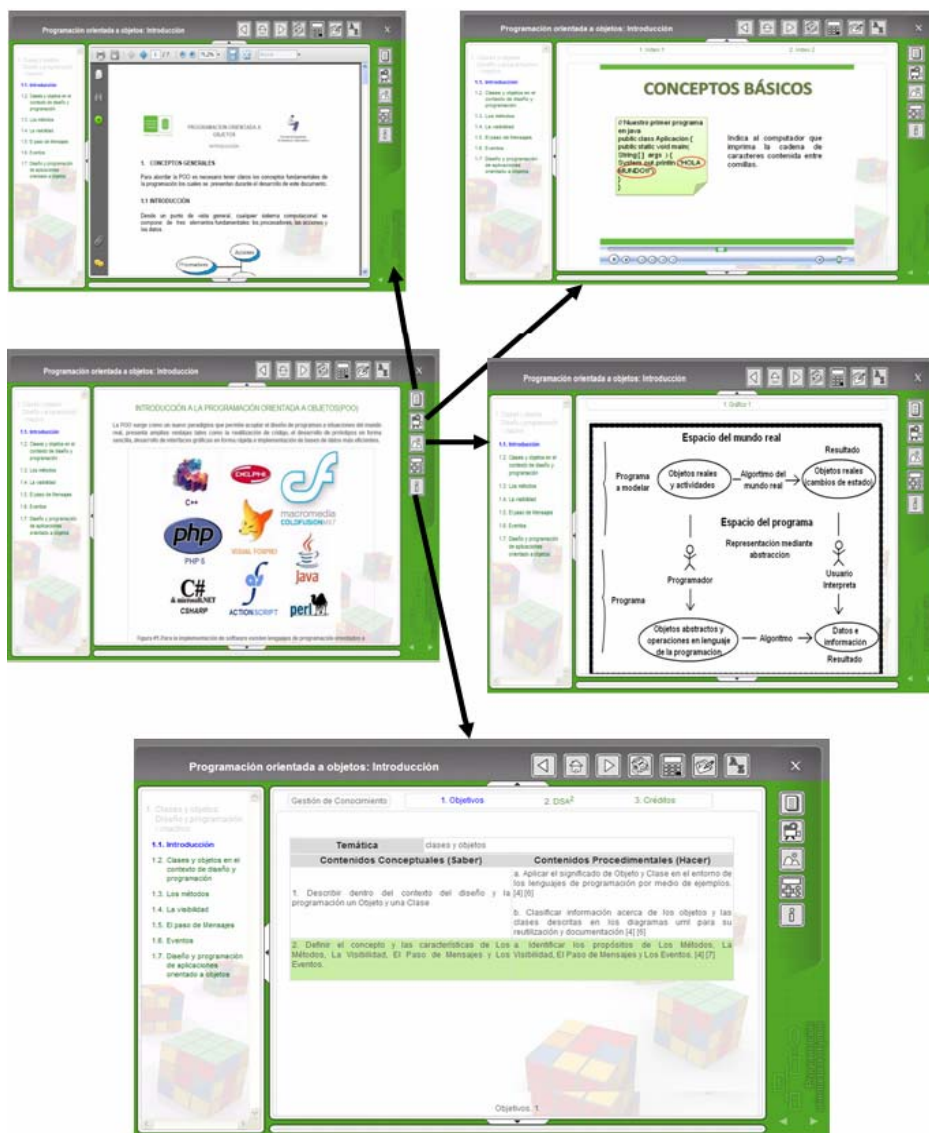


Figura 32. Material en la plantilla para el subtema Introducción

✓ CLASES Y OBJETOS EN EL CONTEXTO DE DISEÑO Y PROGRAMACIÓN

* Objetivos Específico

Comprender los conceptos fundamentales de la programación orientada a objetos para tener una visión de las técnicas utilizadas para el desarrollo de aplicaciones.

Se plantearon los siguientes propósitos:

- *Claridad de los conceptos de objeto, clase, herencia y polimorfismo para su manejo en aplicaciones.
- * Determinar las características de los objetos (métodos, paso de mensajes, visibilidad, eventos) y las de las clases.

Se plantearon las siguientes actividades:

- * Definir el concepto de objeto, clase, herencia, y polimorfismo.
- * Estudiar los pilares de la POO determinando sus características y usos.

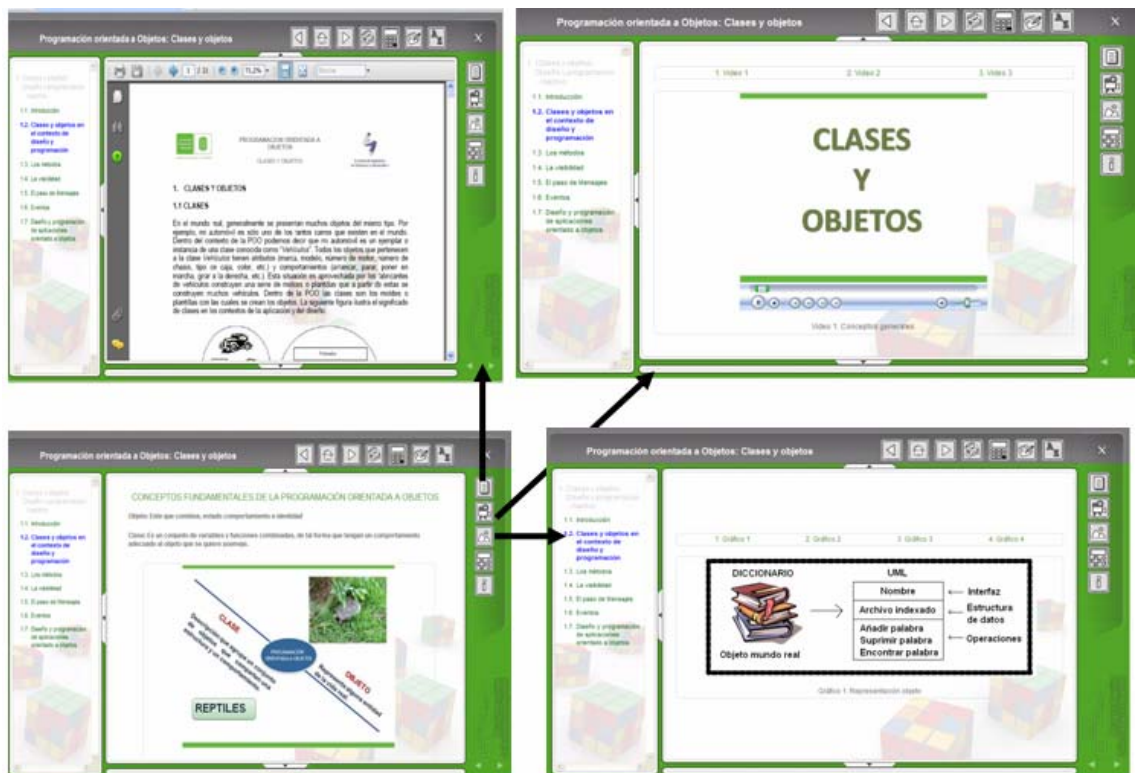


Figura 33. Material en la plantilla para el subtema Clases y Objetos en el contexto de diseño y programación

✓ LOS MÉTODOS

* Objetivos Específico

Determinar las funciones de cada una de las clases utilizadas en aplicaciones software.

Se plantearon los siguientes propósitos:

- * Claridad en el concepto de funciones dentro de una clase.
- * Determinar las instrucciones necesarias para realizar procesos dentro de las clases.

Se plantearon las siguientes actividades:

- * Ilustrar los métodos en diagramas de flujo.
- * Desarrollar casos de uso utilizando métodos en cada una de las clases.

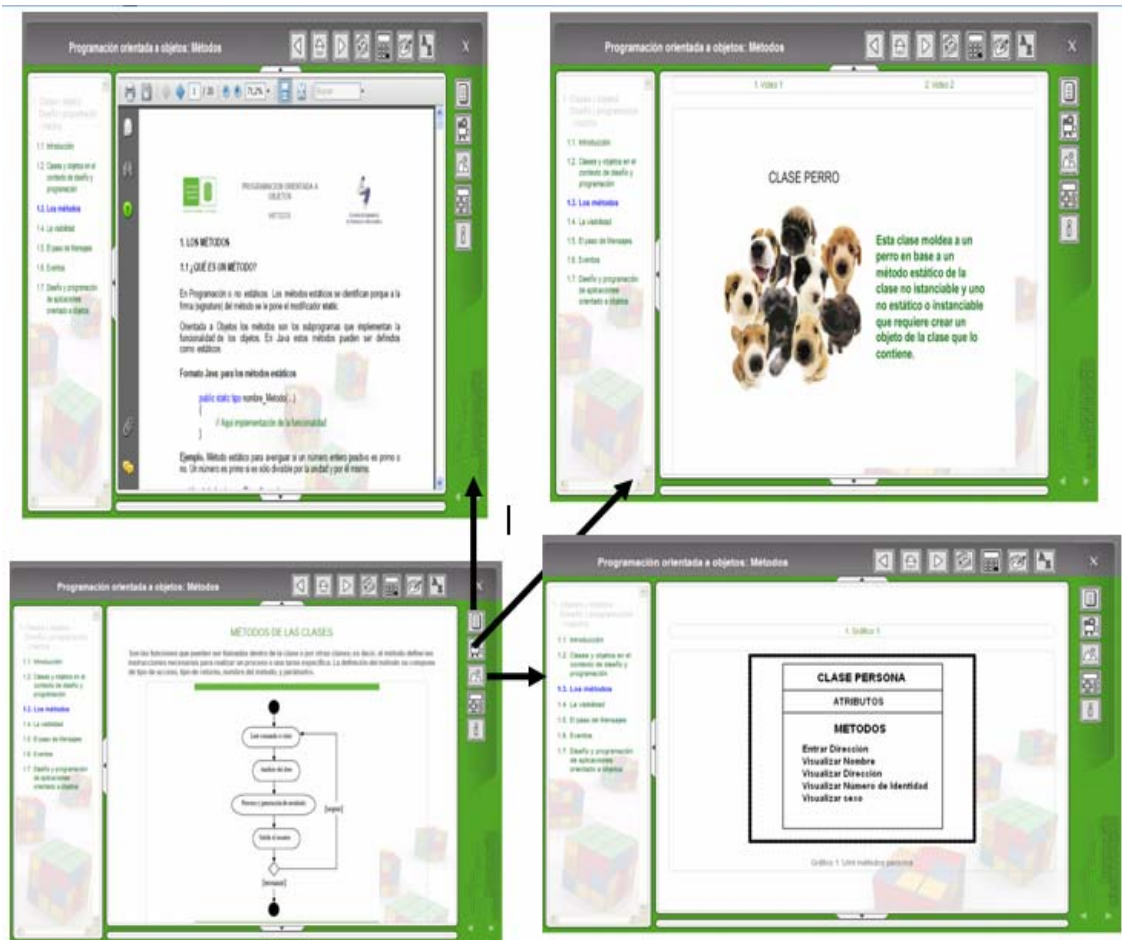


Figura 34. Material en la plantilla para el subtema los Métodos

✓ **LA VISIBILIDAD**

* **Objetivos Especifico**

Definir la forma de acceso a los datos y la disponibilidad de los atributos que componen una aplicación.

Se plantearon los siguientes propósitos:

- * Manejo de la visibilidad para su uso en el desarrollo de aplicaciones.
- * Interpretar la visibilidad en las aplicaciones

Se plantearon las siguientes actividades:

- * Ilustrar la visibilidad en algún lenguaje de POO
- * Desarrollar casos de uso utilizando visibilidad en cada uno de los datos

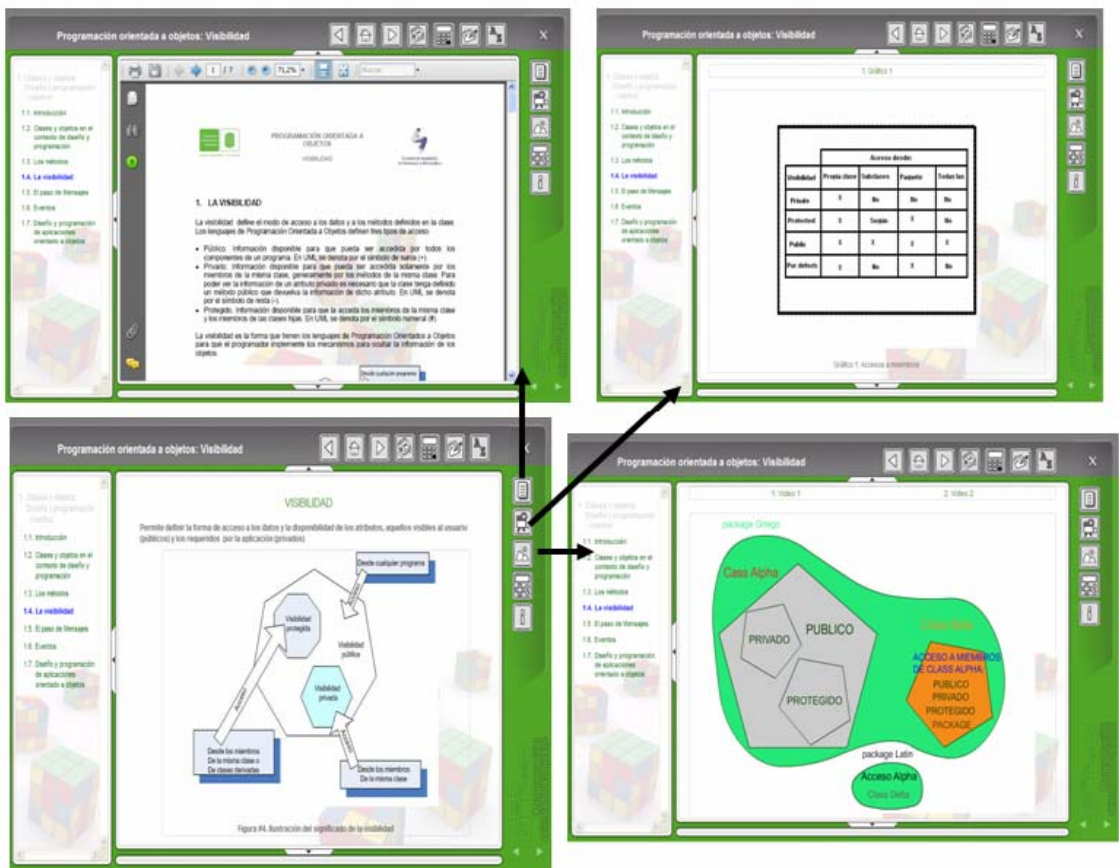


Figura 35. Material en la plantilla para el subtema La Visibilidad

✓ EL PASO DE MENSAJES

* Objetivos Específico

Definir cada una de las llamadas a los métodos mediante el paso de mensajes.

Se plantearon los siguientes propósitos:

- * Manejo de la comunicación entre objetos y sus componentes
- * Interpretar la llamada a los métodos en las aplicaciones

Se plantearon las siguientes actividades:

- * Determinar los componentes de los mensajes
- * Desarrollar casos de uso utilizando la llamada a los métodos.

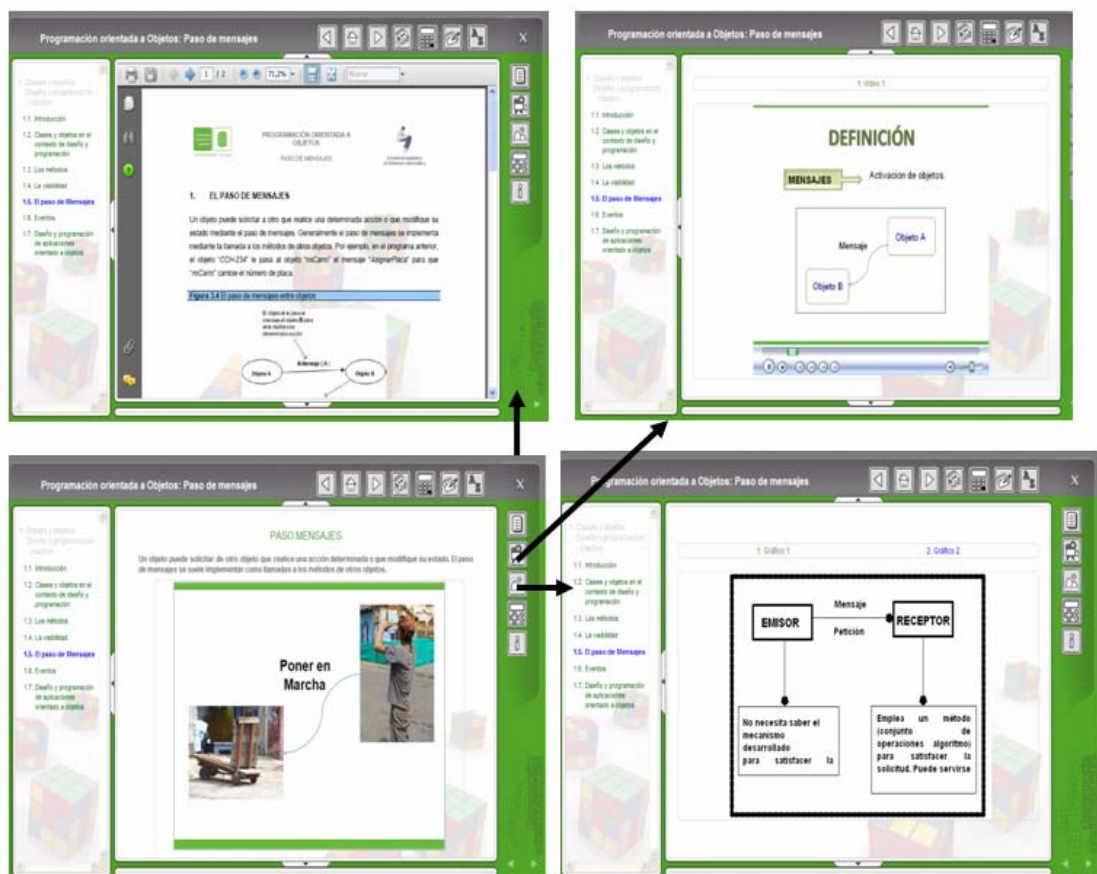


Figura 36. Material en la plantilla para el subtema El Paso de Mensajes

✓ LOS EVENTOS

* Objetivos Específico

Definir cada uno de los cambios de estado de los objetos.

Se plantearon los siguientes propósitos:

- * Manejo de la Interacción entre el usuario y la aplicación
- * Interpretar los estímulos que ejerce un objeto sobre otro

Se plantearon las siguientes actividades:

- * Determinar los cambios de estado de cada uno de los objetos
- * Desarrollar casos de uso utilizando la interacción con el usuario

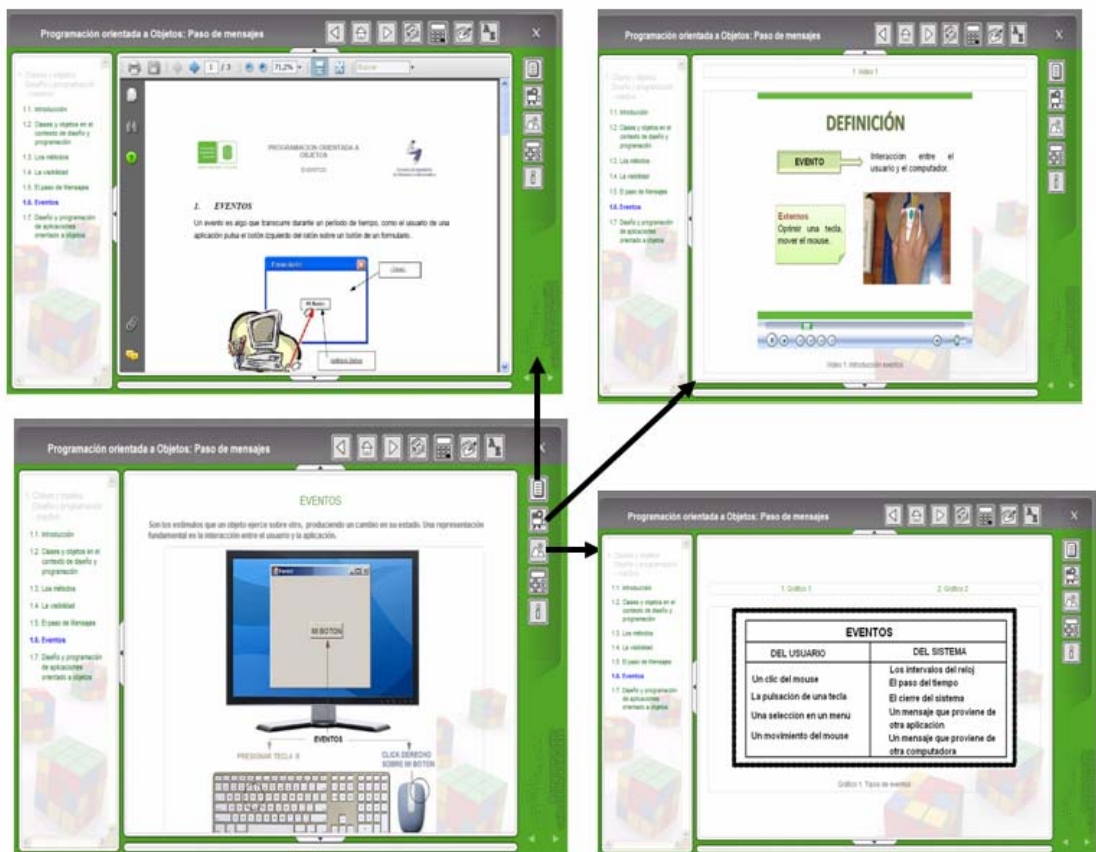


Figura 37. Material en la plantilla para el subtema Eventos

✓ DISEÑO Y PROGRAMACIÓN DE APLICACIONES ORIENTADO A OBJETOS

* Objetivos Específico

Implementar un caso de la vida real mediante software de calidad.

Se plantearon los siguientes propósitos:

- * Contextualizar problemas del mundo real en aplicaciones software
- * Interpretar los requerimientos de una aplicación software

Se plantearon las siguientes actividades:

- * Determinar las entradas y salidas de datos en aplicaciones software
- * Desarrollar una interfaz para una aplicación software basada en un contexto del mundo real.

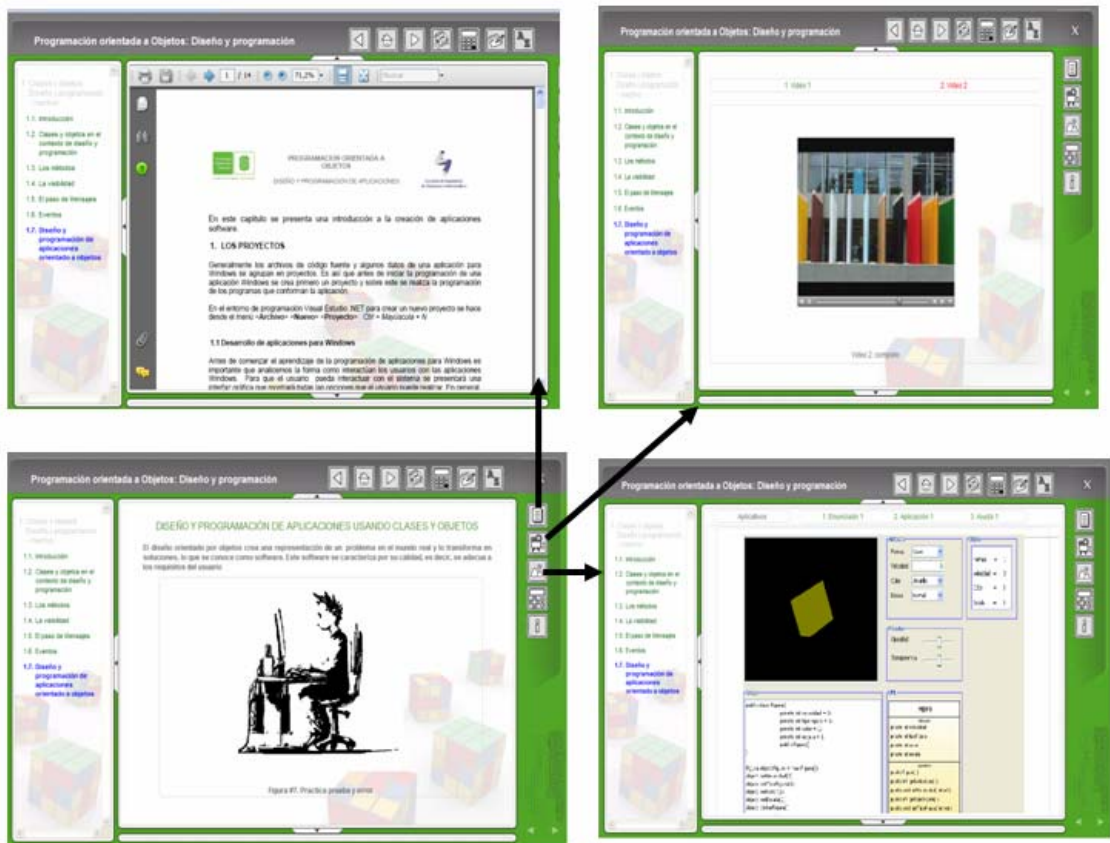


Figura 38. Material en la plantilla para el subtema Diseño y programación de aplicaciones orientado a objetos

5.1.4 Diseño del applet. Para la elaboración de un producto software es necesario determinar los requerimientos tales como: las actividades desarrolladas, la entrada y salida de datos y los resultados, para obtener un producto de calidad.

Se realizó un applet de la temática “Concepto de clases y objetos” el cual permite al usuario interactuar mediante el ingreso de datos y comprobación de un proceso.

La construcción de este objeto se ha basado en un modelo de la vida real: las figuras geométricas, las cuales por sus propiedades como el color, tamaño y forma se acoplan dentro del objetivo de la aplicación.


La construcción del objeto inicia con la implementación de una clase la cual no puede ser modificado por el usuario, dentro ella se crea un objeto, determinado mediante 4 atributos y 2 métodos.

Al realizar cada una de estas acciones el estudiante visualiza y comprueba el proceso que se lleva a cabo:

- La asignación de la memoria al crear un objeto
- La implementación de los atributos en un lenguaje de programación (java)
- El diagrama de la clase en el lenguaje UML

Logrando que el estudiante no solo tenga un marco teórico sino que además pueda complementar las actividades de la asignatura con la parte práctica.

5.1.5 Lenguaje utilizado en el desarrollo del producto software (java). Java es un lenguaje orientado a objetos con un entorno de desarrollo integrado (IDE) netbeans. Su capacidad de portabilidad permite que la aplicación funcione correctamente en cualquier plataforma que tenga una máquina virtual implementada y las librerías necesarias (java3d).



Atributos

Formas:

Velocidad:

Color:

Escala:

Objeto

Formas = 1

Velocidad = 3

Color = 1

Escala = 1

Métodos

Opacidad:

Transparencia:

Código

```

public class Figura{
    private int velocidad = 3;
    private int tipoFigura = 1;
    private int color = 1;
    private int escala = 1;
    public Figura{}
}

Figura objetoFigura = new Figura();
objeto.setVelocidad(3)
objeto.setTipoFigura(1)
objeto.setColor(1)
objeto.setEscala(1)
objeto.pintarFigura()

```

UML

Figura	
<i>Atributos</i>	
private int	velocidad
private int	tipoFigura
private int	color
private int	escala
<i>Operations</i>	
public	Figura()
public int	getVelocidad()
public void	setVelocidad(int val)
public int	getTipoFigura()
public void	setTipoFigura(int val)
public int	getColor()
public void	setColor(int val)
public int	getEscala()
public void	setEscala(int val)
public void	aumentarOpacidad()
public void	disminuirTransparencia()
public void	pintar()

Figura 39. Interfaz del Objeto de Aprendizaje

5.1.6 Instrucciones utilizadas. Las clases y los objetos son los componentes principales de la programación orientada a objetos, para dar inicio a una implementación software es necesario, definir las clases que intervienen y cada uno de los objetos que la componen.

En este applet la clase está predefinida (figuras geométricas), de tal manera que el usuario procede a crear un objeto que pertenezca a esta clase (cubo, cono, cilindro y esfera), creado el objeto, el usuario procede a seleccionar los atributos (predefinidos) que caracterizan al objeto, ellos son:

Velocidad: Entre 1 y 10

Color (Amarillo, Blanco, Rojo, Azul)

Escala (Normal, Doble, Medio)

Definidos los atributos, el usuario procede a seleccionar los eventos propios del objeto:

Opacidad: Determina el grado de oscuridad de la imagen

Transparencia: Visión nítida del objeto.



The image shows a control panel for a Java applet, divided into two sections: 'Atributos' (Attributes) and 'Métodos' (Methods). The 'Atributos' section contains four controls: 'Formas' (Shapes) is a dropdown menu set to 'Cubo'; 'Velocidad' (Velocity) is a text input field containing the number '3'; 'Color' is a dropdown menu set to 'Amarillo'; and 'Escala' (Scale) is a dropdown menu set to 'Normal'. The 'Métodos' section contains two sliders: 'Opacidad' (Opacity) and 'Transparencia' (Transparency), both with green sliders positioned in the middle of their respective horizontal tracks.

Figura 40. Atributos y métodos de los objetos

Cuando el usuario ha creado el objeto con sus atributos y métodos, puede visualizar el espacio en memoria asignado.



Figura 41. Asignación de memoria para el objeto

El usuario, de igual manera, puede apreciar la clase con sus propiedades de visibilidad y cada uno de los cambios efectuados al objeto mediante el lenguaje java.

```
Código
public class Figura{
    private int velocidad = 3;
    private int tipoFigura = 1;
    private int color = 1;
    private int escala = 1;
    public Figura-{}
}

Figura objetoFigura = new Figura();
objeto.setVelocidad(3)
objeto.setTipoFigura(1)
objeto.setColor(1)
objeto.setEscala(1)
objeto.pintarFigura()
```

Este código muestra la implementación de la clase Figura en Java. La clase tiene cuatro atributos privados: velocidad (3), tipoFigura (1), color (1) y escala (1). Se define un constructor público Figura-{}. El código también muestra la creación de un objeto de la clase Figura y la asignación de valores a los atributos mediante los métodos setVelocidad, setTipoFigura, setColor y setEscala. Finalmente, se llama al método pintarFigura().

Figura 42. Implementación de la clase y el objeto en el lenguaje java

Como parte final del objeto de aprendizaje, el usuario puede visualizar la implementación de la clase utilizada mediante el Diagrama de Clases UML.

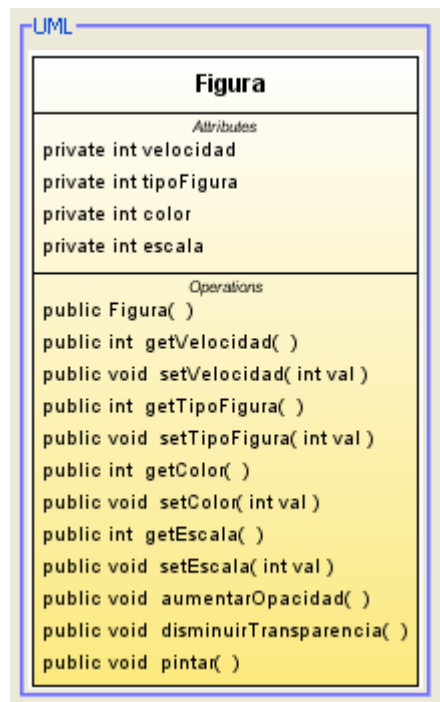


Figura 43. Diagrama de Clase UML

Para facilitar la interacción entre el usuario y la aplicación, el usuario cuenta con una ayuda la cual lo guiará sobre el manejo de la misma, con el fin de cumplir con el objetivo planteado en la temática: concepto de clases y objetos.

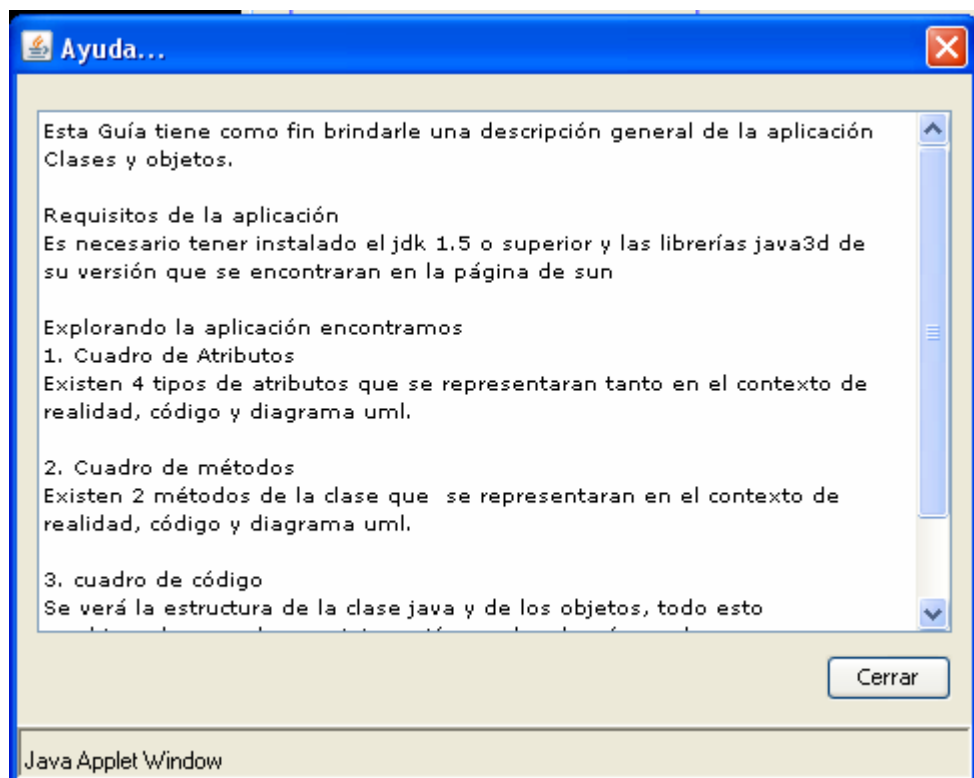


Figura 44. Guía para el manejo del objeto de aprendizaje

5.1.7 Integración y evaluación en la plataforma e-escen@riUIS A continuación se describen cada uno de los elementos que conforman la plataforma.

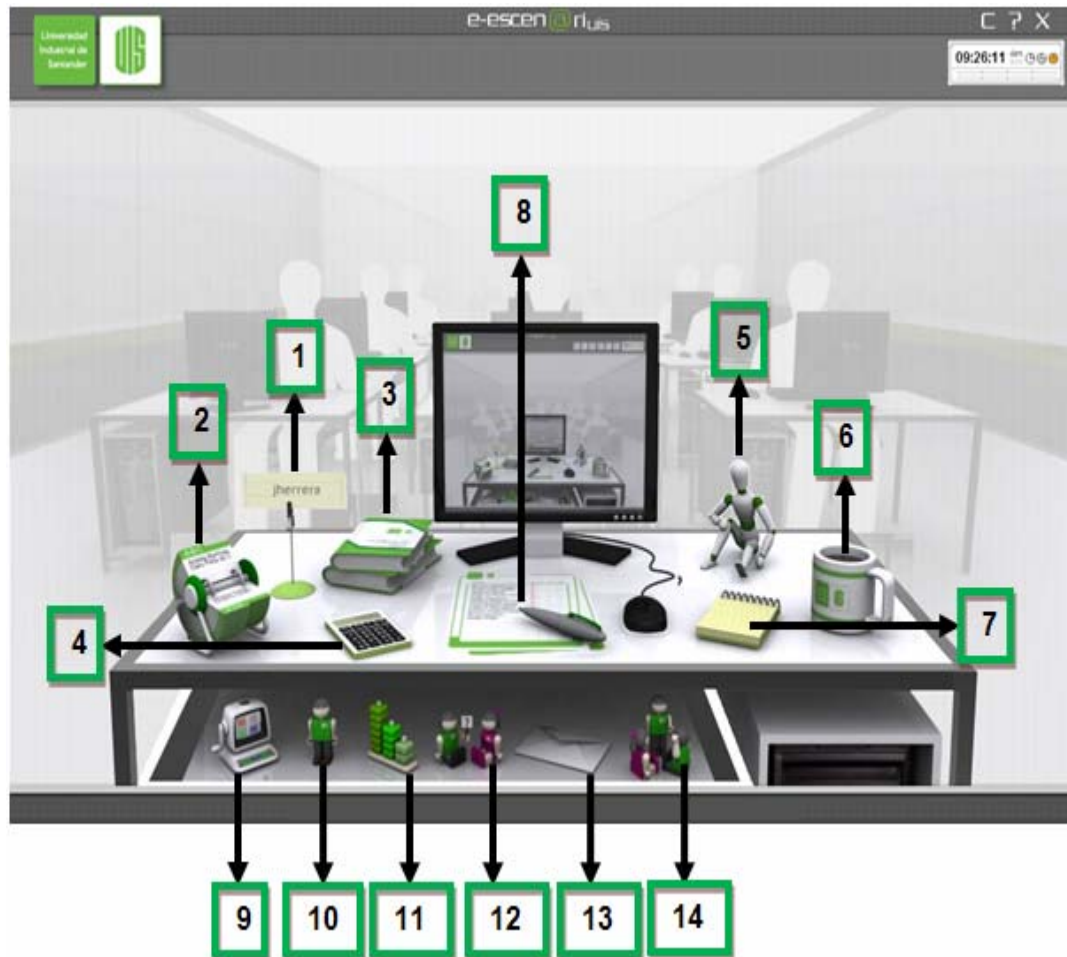


Figura 45. Escritorio de la plataforma e-escen@riUIS

1. Nombre del profesor de la asignatura.
2. Bibliografía.
3. Evaluación (vinculo para la desarrollo de ejercicios).
4. Calculadora.
5. Agente inteligente (es quien brinda un acompañamiento al estudiante en su proceso de enseñanza/aprendizaje).

6. Taza de café (es para representar un descanso por parte del estudiante en la elaboración de ejercicios).

7. Gestor de evaluación.

8. Libreta de Notas.

9. Características del sistema

10. Propiedades del usuario.

11. Estadísticas.

12. Chat.

13. Correo electrónico.

14. Foro.

Al ingresar a la plataforma, sobre el escritorio podemos encontrar el modulo gestor de evaluación, el cual presenta un listado de las asignaturas que el docente enseña (figura 46)

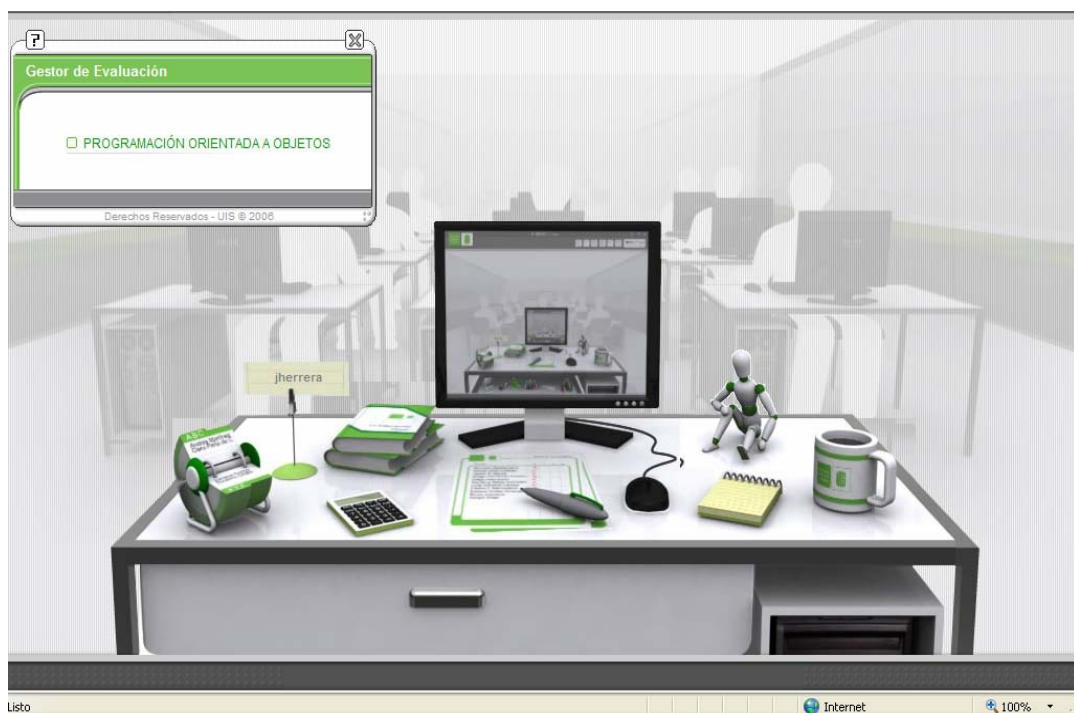


Figura 46. Ventana para la gestión de la evaluación

Para completar el proceso de aprendizaje, los estudiantes desarrollan una evaluación conformada por una serie de ejercicios clasificados en los niveles fácil, medio y difícil y dependiendo del tipo de competencia a evaluar argumentativa: consiste en justificar una idea, interpretativa: capacidad del estudiante para dar respuesta a los problemas que surgen de una situación específica y propositiva: donde el alumno propone hipótesis, soluciona problemas, construye mundos posibles y da alternativas de solución a un contexto específico.

A continuación se muestra y se describe el proceso para la creación de un ejercicio el cual será usado para la evaluación que se realizará sobre la plataforma **e-escen@riUIS**.

En la construcción de un ejercicio se debe tener claro el tema y subtema al que corresponde, además seleccionar el tipo de ejercicio en la ventana de gestión de ejercicios (asociación, completar, ordenar, selección, sopa de letras, cuestionario académico, pregunta abierta).

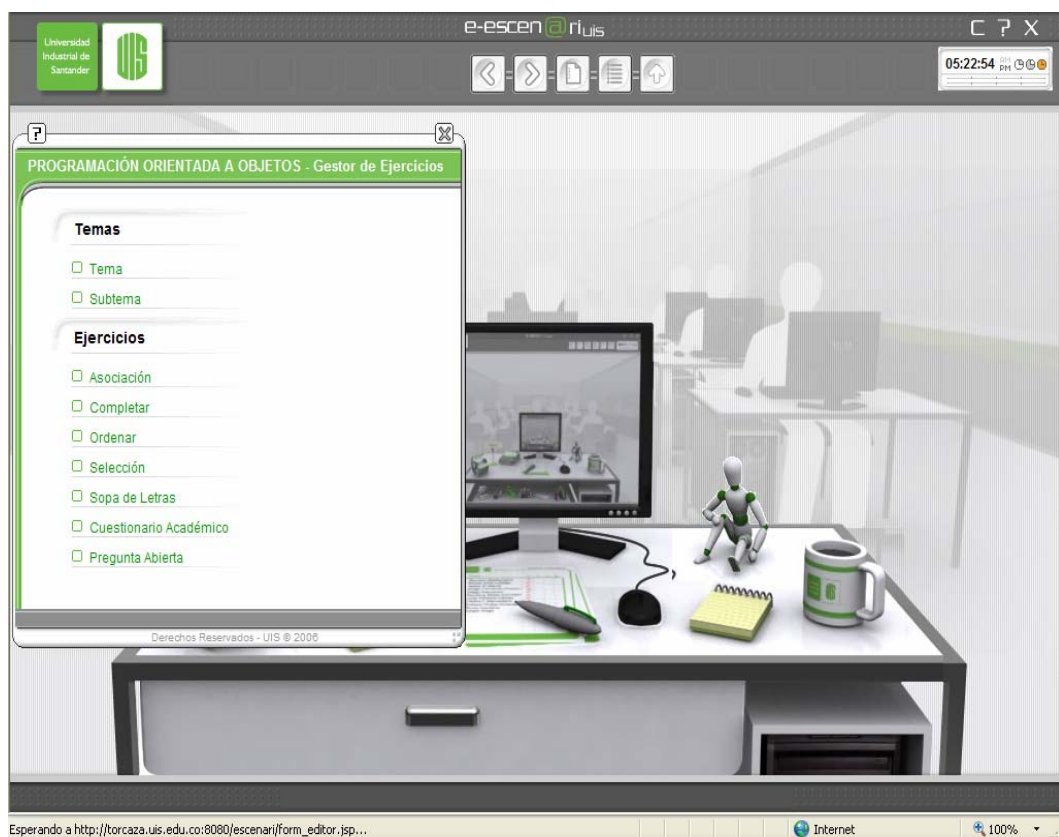


Figura 47. Ventana para la gestión de ejercicios

Después de definir estos parámetros, se selecciona el botón nuevo, que

aparece en la barra de navegación, se debe digitar la información necesaria para la construcción del ejercicio, tal como:

Título: Correspondiente al nombre que identifica al ejercicio.

Descripción: Corresponde a un resumen sobre la temática que se quiere evaluar en el ejercicio.

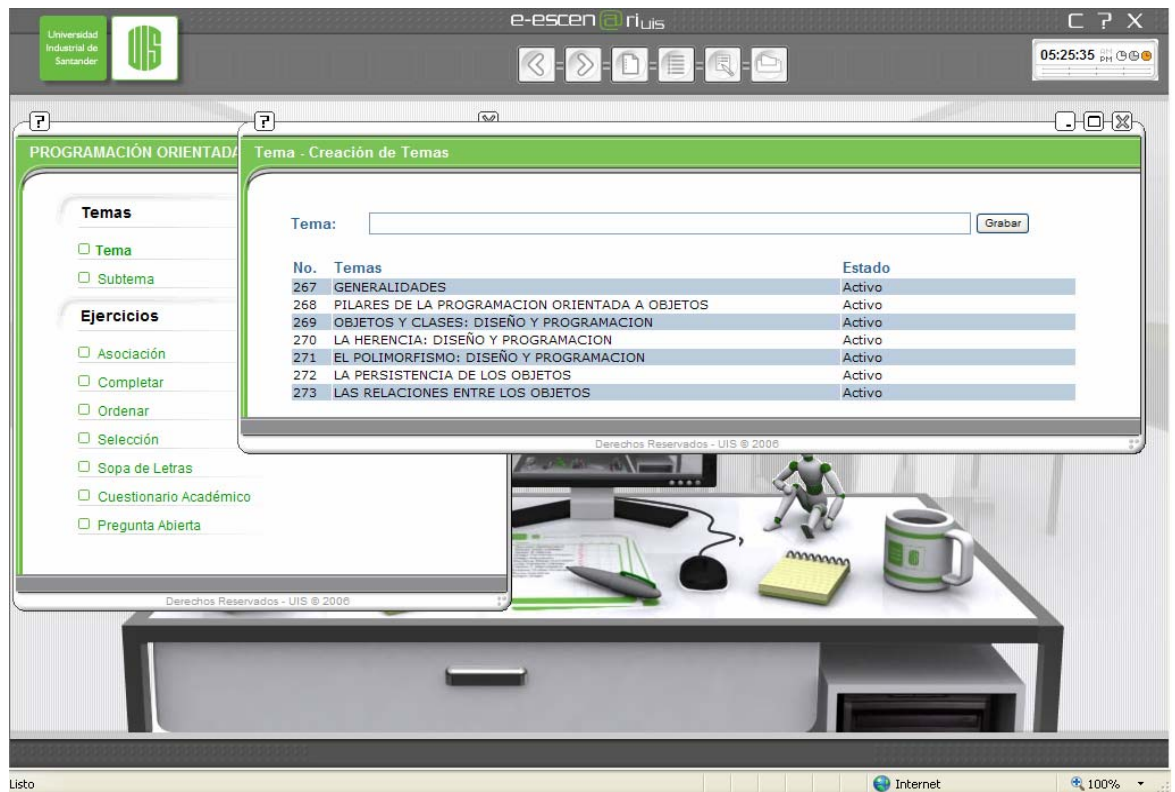


Figura 48. Ventana para la gestión de la evaluación

Se cuenta con algunas cajas de selección, que permiten ver algunas opciones relacionadas con el ejercicio tales como:

Ver Resultado.

Ver Solución.

Grabar Resultado.

Botón de Ayuda.

Ver Retroalimentación.

Aleatoriedad.

En cuanto a su configuración es importante definir:

Número de Ejecuciones.

Fórmula para el cálculo de la nota de la evaluación del ejercicio.

Número de Asociaciones.

Número de Asociaciones A Mostrar.

Selecciones el Tipo de Asociación.
Tiempo Máximo.

Una vez se llene la información concerniente a cada ejercicio se da por construido el ejercicio.

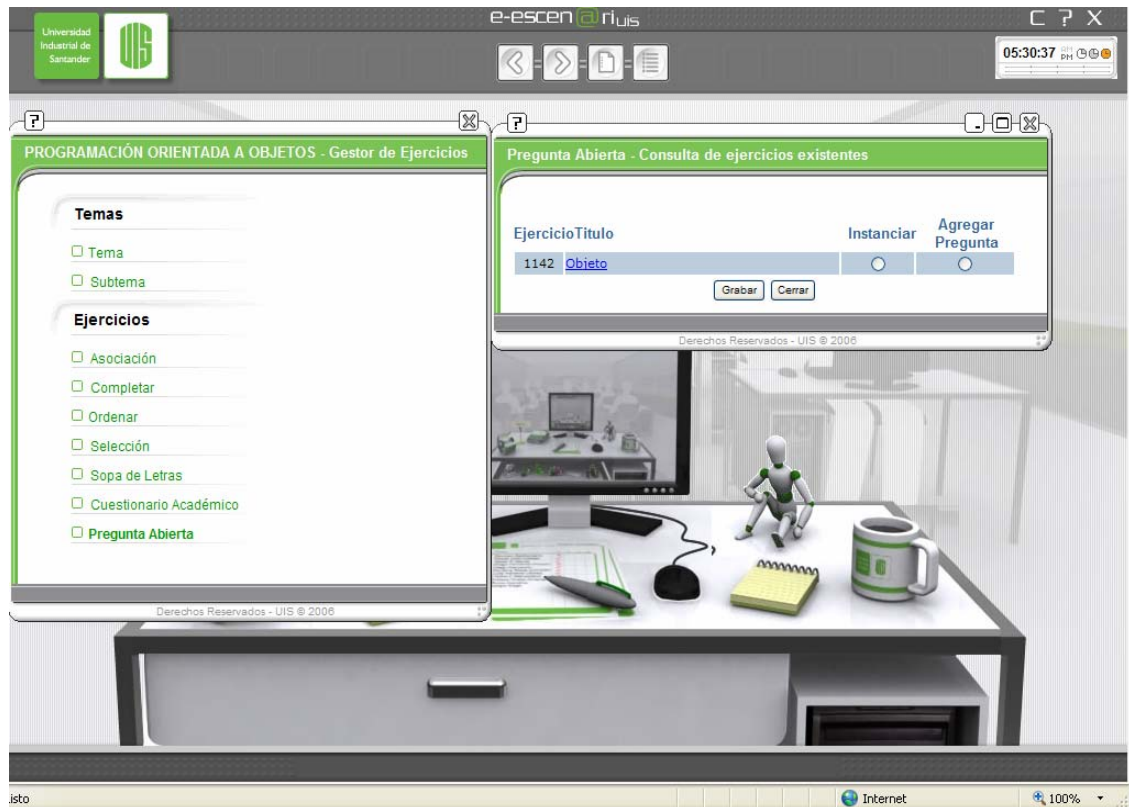


Figura 49. Ejercicios construidos en la temática concepto de clases y objetos

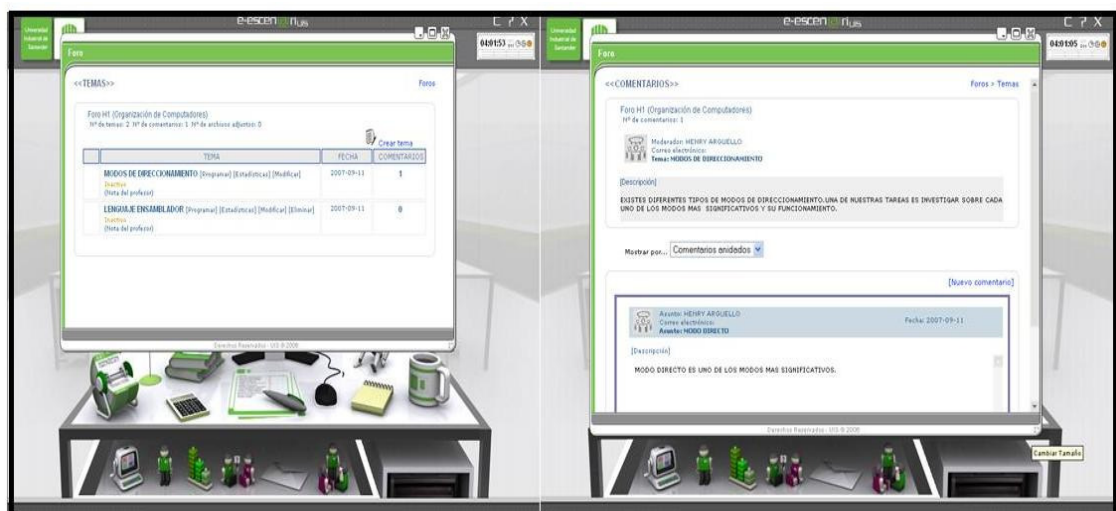


Figura 50. Actividades de Trabajo colaborativo

5.2 ENCAPSULAMIENTO DEL OBJETO DE APRENDIZAJE

5.2.1 Generación de los metadatos y encapsulamiento del objeto. Para la generación y encapsulamiento del objeto de aprendizaje, se utilizó la herramienta de libre distribución llamada RELOAD²⁰.

El editor de Reload es una aplicación Java estable y funcional que puede ejecutarse en cualquier plataforma capaz de ejecutar aplicaciones Java, permite la edición de los metadatos y el encapsulado del material didáctico que conforma el objeto de aprendizaje, siguiendo el estándar SCORM, cumpliendo con características de usabilidad, interoperabilidad y actualización; permitiendo "transportar" contenidos educativos de un sistema formativo a otro.

* Requisitos del Sistema.

Para ejecutar el Editor de Reload en el PC se debe tener al menos: Procesador Intel Pentium III (o equivalente), a 800 MHz.

256 Mb RAM.

Microsoft Windows 95, 98, Me Windows NT4.0, Windows 2000 ó XP.

Un navegador para visualizar los Paquetes de Contenidos.

Ante la expansión de iniciativas destinadas a la generación de contenidos educativos se plantea la necesidad de establecer estándares de generación y catalogación de estos contenidos que garanticen su aprovechamiento por parte de la comunidad educativa.²¹

Con la aplicación de estándares de e-learning se persiguen los siguientes requerimientos funcionales:

- Accesibilidad desde diferentes sitios a través del uso de metadata y estándares de empaquetamiento.
- Adaptabilidad a los requisitos específicos del usuario final (persona u organización).
- Ahorro al incrementar la eficiencia y productividad del aprendizaje reduciendo tiempos y costes de creación y explotación.
- Durabilidad frente a evoluciones tecnológicas sin necesidad de rediseñar o reconfigurar los recursos.
- Interoperabilidad entre diferentes herramientas o plataformas.
- Reusabilidad de los recursos en diferentes aplicaciones y contextos.

²⁰ <http://www.reload.ac.uk>. Software de libre distribución para la generación y encapsulamiento del Objeto de Aprendizaje.

²¹ Tomado de la guía elaborada por el grupo de investigación y desarrollo CENTIC

El modelo de uso de objetos de aprendizaje para e-escen@riUIS requiere un conjunto de estándares de etiquetado y empaquetamiento de los contenidos.

Para dar inicio al empaquetamiento se debe tener el material del objeto de aprendizaje, herramientas multimedia, animaciones, applet de Java, los documentos PDF, videos, etc. Cada formato ubicado en las carpetas respectivas.

Creación de un Paquete SCORM:

Seleccionar el icono “Nuevo”.

Seleccionar la opción “Paquete SCORM”.

En el cuadro “seleccionar carpeta para un nuevo paquete”, seleccionar la carpeta en la que se creará el paquete y pulsar el botón “Select”

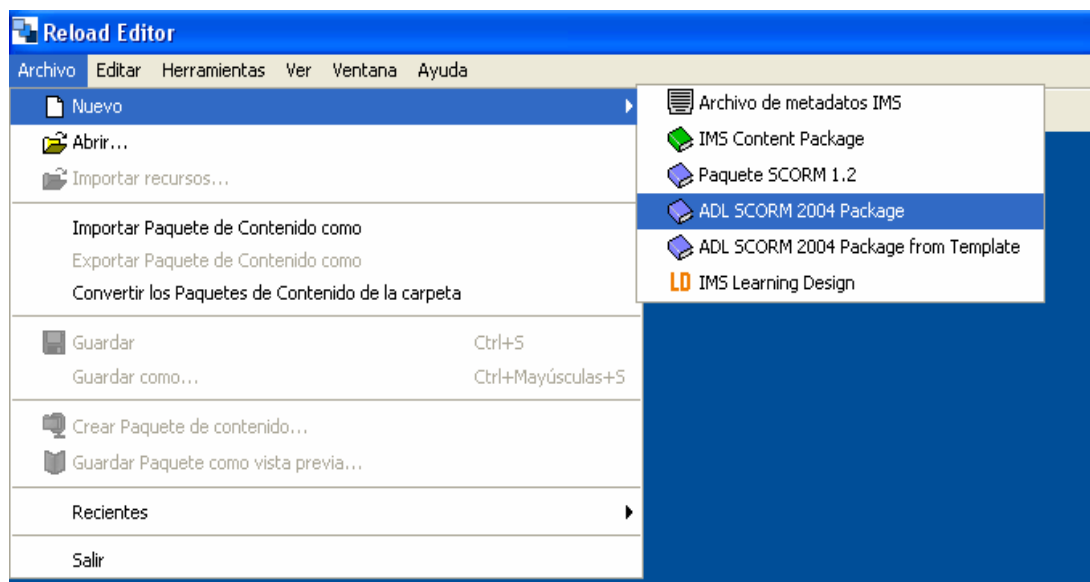


Figura 51. Creación de un paquete SCORM

La herramienta RELOAD, brinda un entorno de trabajo en el cual se encuentran los paneles de recursos, manifiesto y atributos los cuales describiremos a continuación.

- Panel de recursos: muestra de forma organizada el material a encapsular.
- Panel del manifiesto: es el espacio donde se encuentran los metadatos, las organizaciones y los recursos.
- Panel de atributos: allí se puede visualizar la información del elemento seleccionado, y donde se puede modificar su contenido

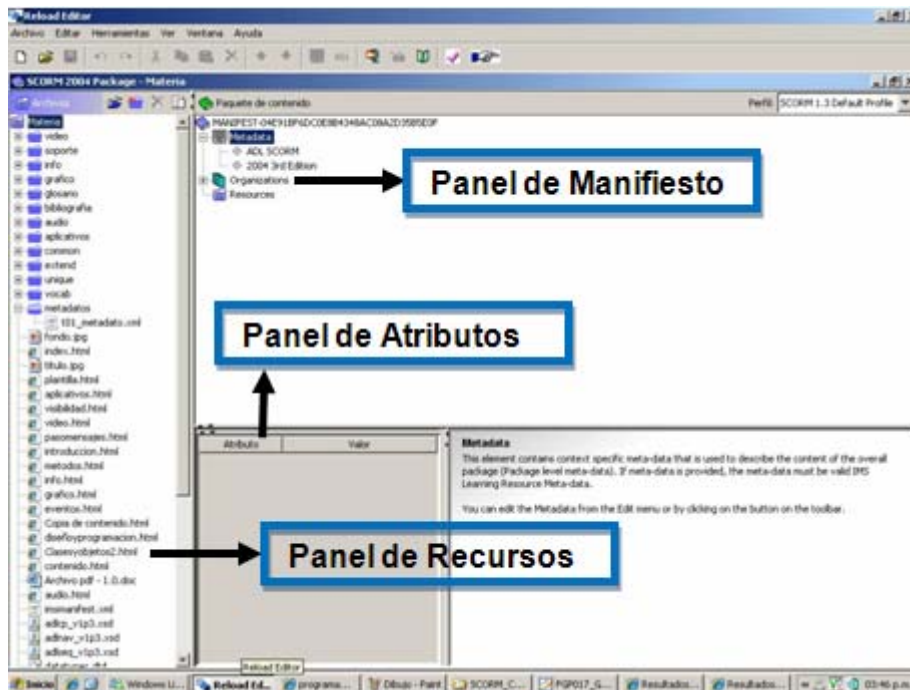


Figura 52. Escritorio de trabajo de la herramienta RELOAD

* Las acciones a realizar son:

Seleccionar el icono “Nuevo carpeta”.

Se crea una carpeta llamada metadato.

En el mismo cuadro “seleccionar aceptar para crear el archivo y trabajar ahí el IMSMANIFIEST”.

Se selecciona el icono actualizar para que la carpeta .XML quede en el panel de archivo.

Los metadatos son los datos informativos y estandarizados que tiene que contener todo paquete SCORM para poder ser utilizado por los diferentes LMS (Learning Management System) o entornos virtuales de enseñanza/aprendizaje). Por lo tanto el material encapsulado tiene características como nombre, versión, autor, palabras claves, etc. los cuales se diligencian a través de un formulario el cual hace parte de la herramienta RELOAD.

* Las acciones a realizar para agregar los metadatos:

Seleccionar el icono “Manifest” En el panel de manifiesto.

Seleccionar editar METADATOS en el panel manifiesto.

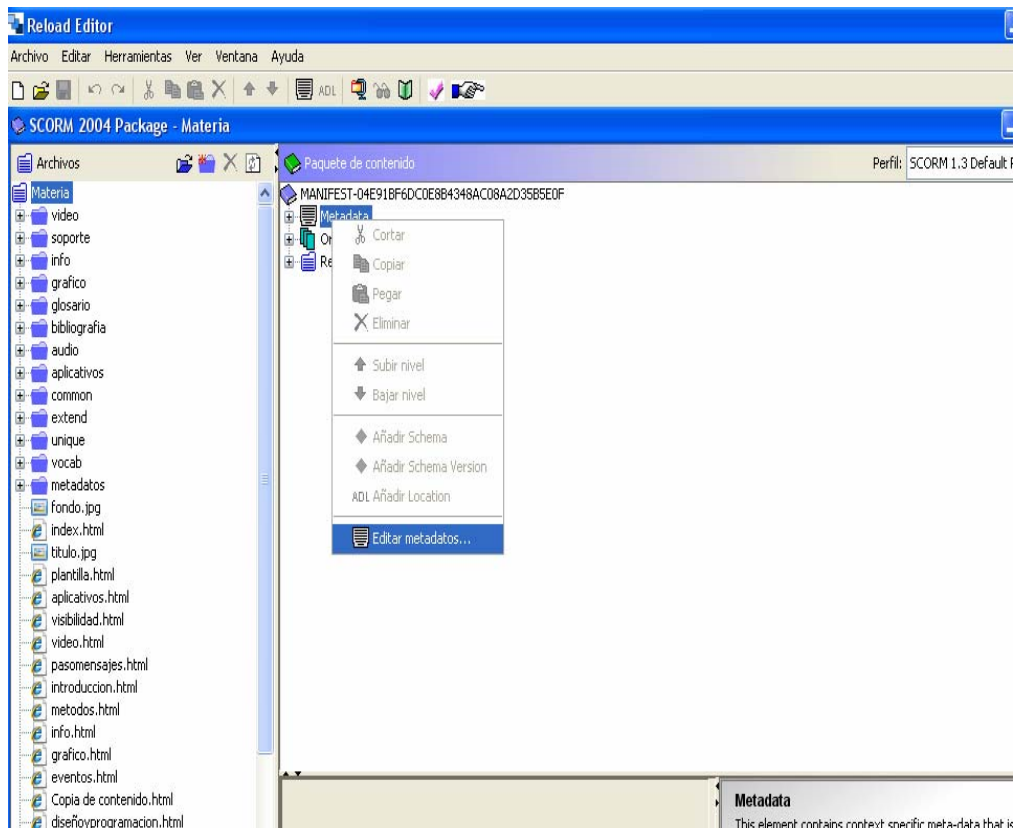


Figura 53. Edición de los metadatos

Esta lista de elementos requeridos se puede aplicar para cualquiera de los componentes del modelo de contenidos de SCORM (Asset, SCO, Actividad, Organización de contenidos, Objeto de Aprendizaje). Para el caso de la Biblioteca Digital de Recursos Didácticos (BDRD), aquí se presenta los elementos obligatorios y opcionales que deben ser aplicados en el empaquetamiento del objeto de aprendizaje estos se ingresan a su vez mediante la vista formulario, vista esquema o vista formulario completo.

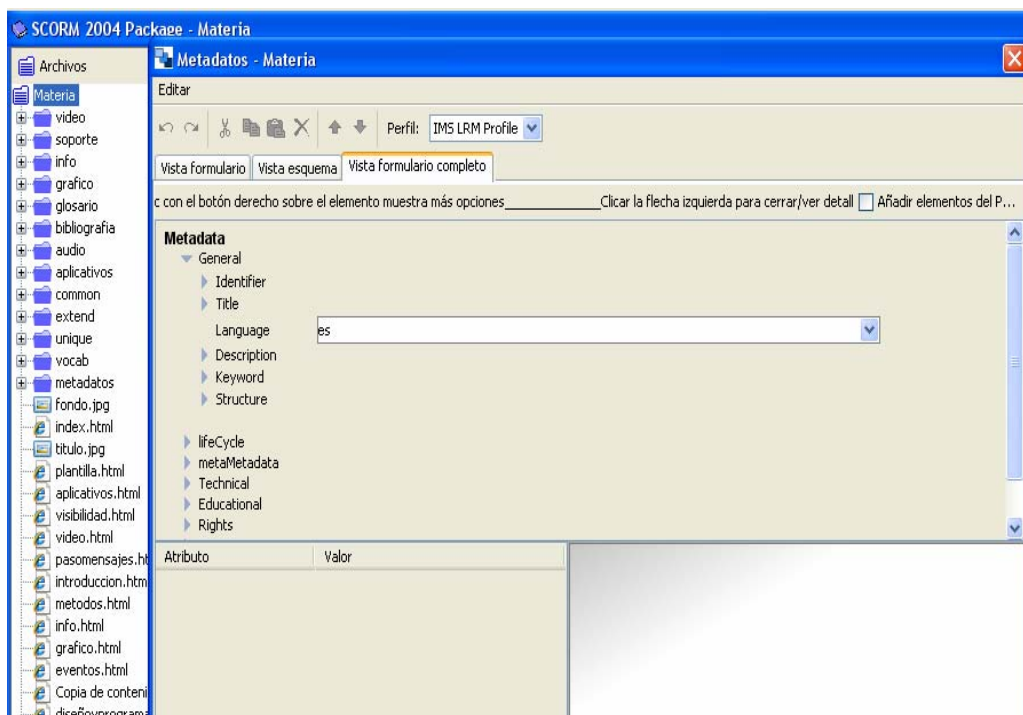


Figura 54. Introducción en un LMS

El formato definido para e-escen@riUIS utiliza las 9 categorías de metadatos XML propuesta por el LOM (Learning Object Metadata):

1. La categoría General (general) agrupa la información general que describe el objeto de aprendizaje de manera global.
2. La categoría Ciclo de Vida (lifeCycle) agrupa las características relacionadas con la historia y el estado actual del objeto de aprendizaje, y aquellas que le han afectado durante su evolución.
3. La categoría Meta-Metadatos (metaMetadata) agrupa la información sobre la propia instancia de metadatos, (en lugar del objeto de aprendizaje descrito por la instancia de metadatos).
4. La categoría Técnica (technical) agrupa los requerimientos y características técnicas del objeto de aprendizaje.
5. La categoría Uso Educativo (educational) agrupa las características educativas y pedagógicas del objeto de aprendizaje.
6. La categoría Derechos (rights) agrupa los derechos de propiedad intelectual y las condiciones para el uso del objeto de aprendizaje.
7. La categoría Relación (relation) agrupa las características que

definen la relación entre este objeto de aprendizaje y otros objetos relacionados.

8. La categoría Anotación (annotation) permite incluir comentarios sobre el uso educativo del objeto e información sobre cuándo y por quién fueron creados dichos comentarios.

9. La categoría Clasificación (classification) describe este objeto de aprendizaje en relación a un determinado sistema de clasificación.

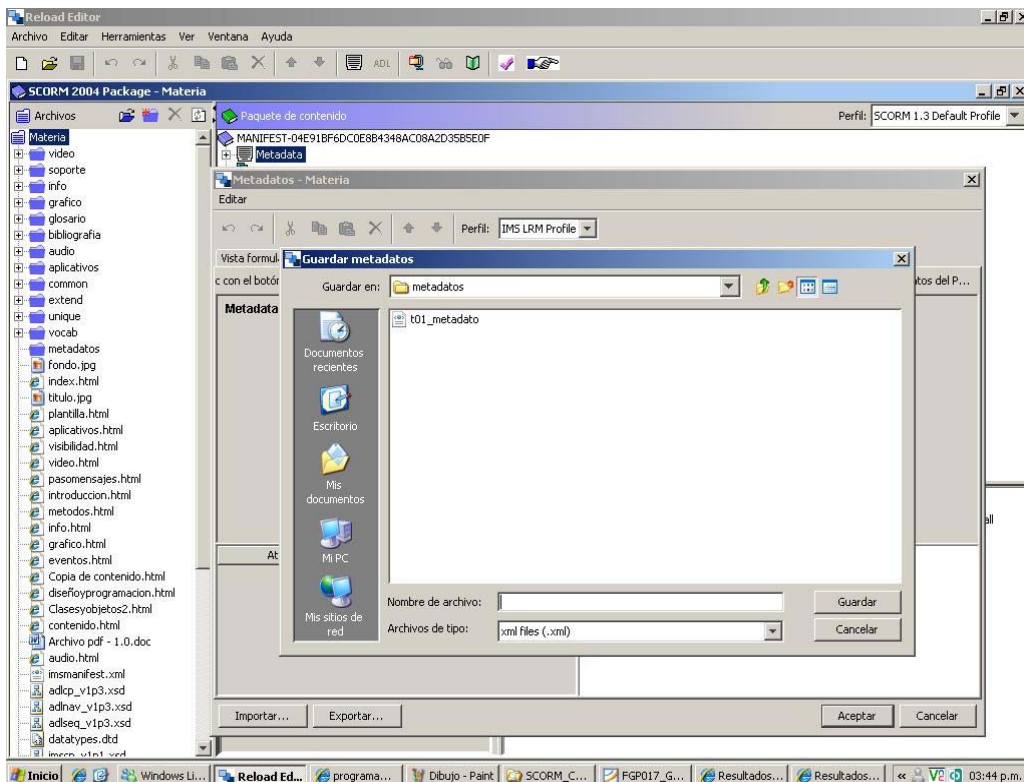


Figura 55. Creación de la carpeta metadatos

Para ir dando forma a la estructura de aprendizaje que se le quiere dar al paquete se crea, una organización. Un paquete puede tener muchas estructuras, denominadas organizaciones.

La organización del paquete vendrá dada por la secuencia de los contenidos que se van añadiendo. Arrastrando y soltando cada elemento desde la lista de la izquierda hasta el nombre de la organización ubicando en cada una de ellas, el material correspondiente a cada organización creada.

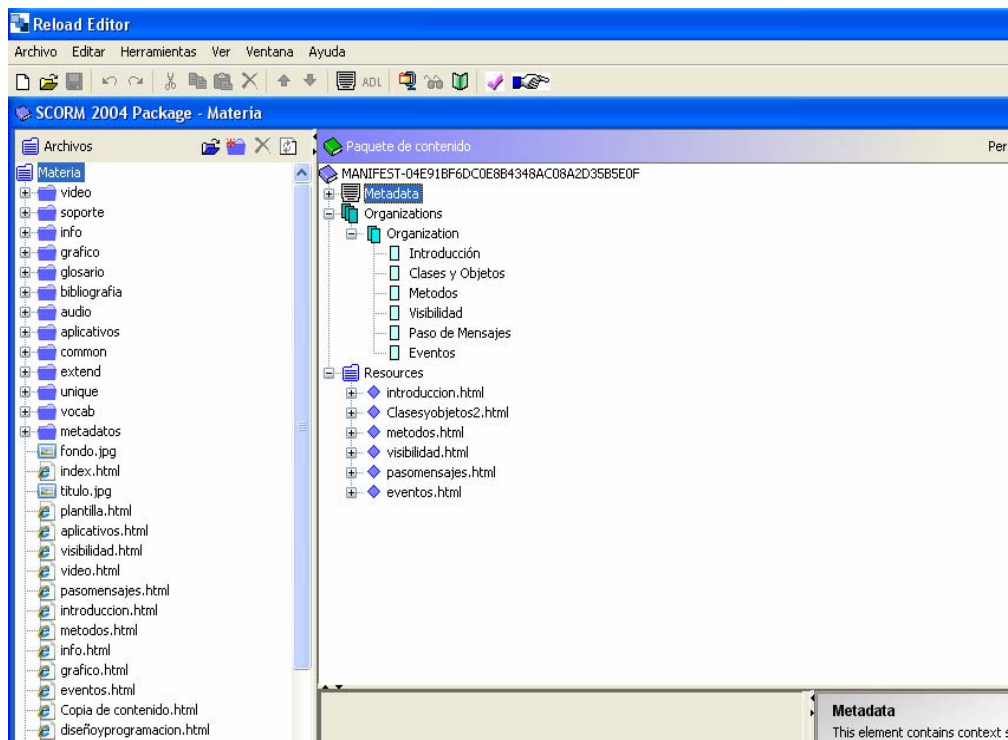


Figura 56. Añadir la Organización a la Estructura del Objeto de Aprendizaje

- Archivos o carpetas del panel de recursos a los recursos del panel del manifiesto.
- Archivos o carpetas del panel de recursos a los ítems de las organizaciones del panel del manifiesto.
- Carpetas del panel de recursos para convertirse en Organizaciones del panel del manifiesto
- Recursos del panel de manifiesto para convertirse en ítems del mismo.
- Objetos del escritorio de nuestro PC a una carpeta del panel de recursos.

Estos se pueden reordenar dentro de la organización. Se hace clic sobre el nombre con el botón derecho del ratón y en el menú desplegable se selecciona "Subir Nivel" ó "Bajar Nivel". De la misma manera se puede modificar el nombre del ítem en la ventana de edición de la zona inferior, al ser seleccionado.

Una vez que se han añadido los contenidos a la organización u organizaciones del paquete y salvados los cambios, se puede previsualizar en una ventana del navegador. Para esto, se da clic en la barra de herramientas en el icono "Vista previa paquete" y se abrirá una nueva ventana con los contenidos del mismo.

Para guardar el contenido del paquete en un archivo comprimido .zip, se hace clic en el icono "Crear paquete de contenido". En el cuadro de diálogo que aparece, debe elegir un nombre para el paquete (incluyendo la extensión.zip) y se elige un directorio.

Se obtiene un paquete SCORM de objetos de aprendizaje preparado para ser distribuido por la red, intercambiado o utilizado en un entorno virtual LMS, como por ejemplo e-escen@riUIS, Moodle, Blackboard, etc.

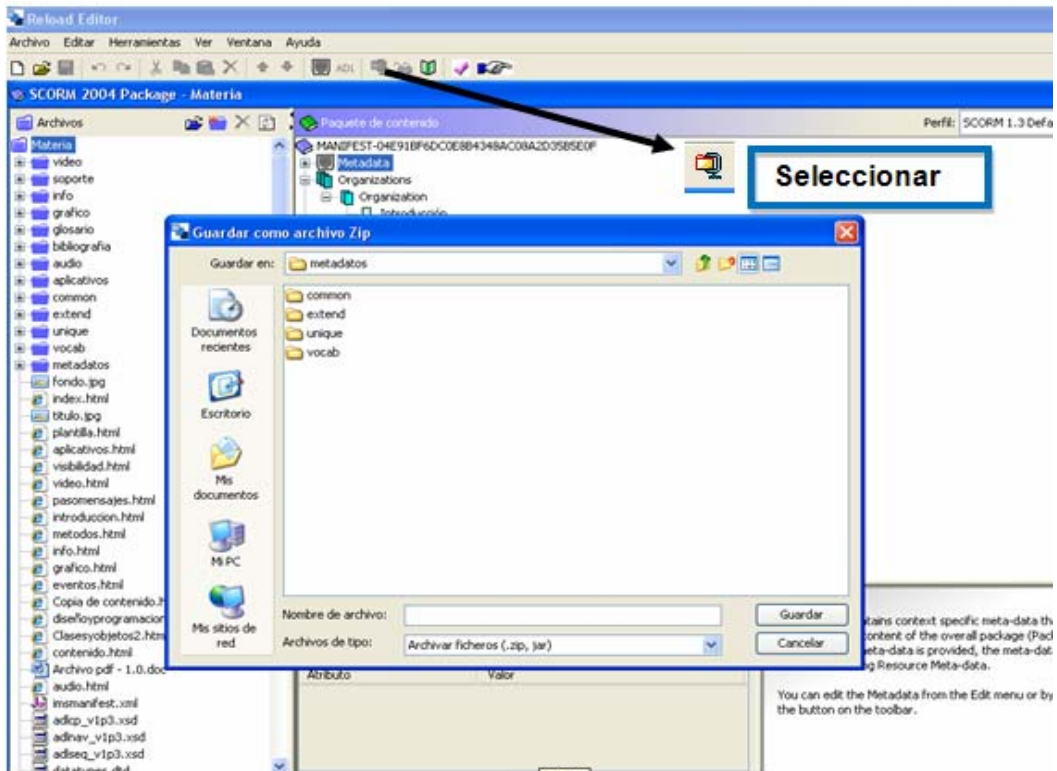


Figura 57. Creación de un paquete con RELOAD.

6. PORTAL DEL PROFESOR

El portal del profesor fue creado con la finalidad de dar a conocer la labor docente a través de la web, de esta forma, crear una cultura en la red que facilite la comunicación entre docente-estudiante.

A continuación se explicará de forma detallada cada uno de los vínculos que componen el portal web del profesor Jorge Herrera Castillo referente a la asignatura Programación Orientada a Objetos.

Para ingresar al portal (ver figura 61) se realiza a través del siguiente enlace: <http://gavilan.uis.edu.co/~jherrera>

En la página principal del portal se encuentran datos importantes del docente, y en la parte superior se encuentran enlaces que permiten ver la labor del docente de forma detallada los cuales se describen a continuación:



Figura 58. Página principal del portal del profesor

1. Inicio

Este enlace sirve para ir a la página principal del portal cuando se encuentre en otro enlace; esta página contiene el nombre de la Escuela a la cual pertenece el profesor, para el caso Ingeniería de Sistemas e Informática, el correo electrónico que le proporciona al universidad al profesor y por último la dirección del campus universitario donde el labora.

2. Curriculum

Este enlace muestra el currículum del profesor en formato PDF, podemos ver los datos personales, estudios secundarios, universitarios, especializaciones y maestrías, áreas de investigación, grupos de investigación, publicaciones, trabajos dirigidos o en marcha, seminarios y cursos dictados, experiencia laboral.

3. Docencia

Portal del Profesor Jorge Herrera Castillo - UIS-DSI - Windows Internet Explorer
http://gavilan.uis.edu.co/~jherrera/index_general.htm

Portal del profesor
Escuela de Ingeniería de Sistemas e Informática 11:25:38

Inicio Curriculum Docencia Investigación Extensión Administración Enlaces de Interés Noticias Salir

Jorge Herrera Castillo
Ingeniero de Sistemas
Magister en Informática
jherrera@uis.edu.co
Escuela de Ingeniería de Sistemas e Informática
Teléfono: 57 7 6344000 ext. 2341
Fax: 57 7 6349042
Universidad Industrial de Santander
Carrera 27 Calle 9
Apartado de correos 678
Bucaramanga, COLOMBIA

DOCENCIA

- 1 [Programación Orientada a Objetos](#)
- 2 [Fundamentos de Programación](#)

e-escen a ruis
Usuario:
Contraseña:
Idioma: Español

Dirección:

Resolución de pantalla recomendada 1024 x 768 pixels
© Copyright 2005 Universidad Industrial de Santander UIS Bucaramanga Colombia - Todos los derechos reservados

Figura 59. Enlaces a Docencia.

Al dar clic en el enlace, muestra en la parte izquierda de la plantilla las asignaturas que el profesor dicta, el nombre de cada asignatura contiene un enlace que permite acceder a la documentación que allí se encuentra, tales como: contenido de la materia, plan de estudios (escuela de Ingeniería de sistemas e Informática), los nombres, fotos y correos electrónicos de cada uno

de los correspondientes grupos de estudiantes de los cuales el profesor tiene a cargo.

4. Investigación

Este enlace permite ver las investigaciones que ha realizado el docente en el transcurso de su proceso de formación laboral y profesional, mediante un enlace se puede acceder a ellos los cuales se encuentran en formato PDF.

5. Extensión

Aquí se encuentra la información referente a cursos adicionales, grupos de investigación y actividades de formación académica realizadas por el docente que son complemento a su labor.

6. Administración

En esta página se dan a conocer los cargos administrativos que el docente ha ejercido o que actualmente se encuentra ejerciendo dentro de la Universidad Industrial de Santander.

7. Enlaces de interés

En esta parte del portal la información que se presenta corresponde a sitios que el docente considera necesarios para complementar las temáticas relativas a la asignatura Programación Orientada a Objetos.

8. Noticias

En esta página se dan a conocer constantemente noticias que el docente considere necesarias para las actividades diarias de cada una de las asignaturas que dicte, tales como información de previos o trabajos, fechas de evaluación de la asignatura, congresos, o aquella noticia que sea de interés para los alumnos respecto a la parte académica no necesariamente de la asignatura etc.

9. Salir

Una vez concluida la navegación por el portal del profesor este permite salir totalmente de la página.

10. Navegador

La ventana dispone de un navegador que permite tanto al alumno como al profesor acceder a otros sitios relacionados con el proceso de enseñanza/aprendizaje de la materia sin necesidad de salir del portal, con el fin de no desviar la atención del estudiante logrando que cumpla con los objetivos propuestos en cada tema tratado.

CONCLUSIONES

El análisis funcional es una metodología para el desarrollo de competencias laborales, al adecuarlo al contexto educativo se asegura la integración de conocimientos, habilidades, prácticas y valores del estudiante, con la formación necesaria para desempeñarse con éxito en campos específicos relacionados con su profesión.

Mediante el uso de las tecnologías de información y comunicación se llevó a cabo el diseño y desarrollo de un objeto de aprendizaje de la temática “Concepto de clases y objetos” implementado bajo los estándares SCORM, que cumplen con los parámetros exigidos por la plataforma institucional *e-escena@riuis* y está en concordancia con los lineamientos propuestos por la metodología APROA.

El funcionamiento del objeto de aprendizaje “Concepto de Clases y Objetos” permite la profundización de los contenidos en diferentes formatos digitales los cuales apoyan el proceso de enseñanza/aprendizaje teniendo en cuenta el modelo de FLSM aplicado en el desarrollo Instruccional de la asignatura Programación Orientada a Objetos.

El portal del profesor aporta material que complementa el proceso de enseñanza/aprendizaje de la asignatura “Programación Orientada a objetos” permitiendo realizar un seguimiento a cada una de las actividades planteadas mediante la comunicación entre el docente y el estudiante.

RECOMENDACIONES

Es muy importante en el proceso de enseñanza/aprendizaje dar continuidad en el desarrollo de objetos de aprendizaje que abarquen el contenido de toda la asignatura, puesto que este proyecto solamente presenta este proceso para la temática de clases y objetos.

Es fundamental incentivar la cultura de trabajo en la red a través de la implantación y actualización del portal del profesor, ya que este le da visibilidad internacional al profesor UIS, y brinda soporte a los estudiantes que tienen acceso a la información de los contenidos referentes a la asignatura permitiendo profundizar en los temas tratados mediante los diferentes recursos multimedia completando el proceso de enseñanza/aprendizaje.

Para futuras propuestas de diseño curricular basado en competencias conviene realizar un enfoque más profundo en torno al apoyo de varios expertos temáticos relacionados con la asignatura que se está ilustrando y de esta forma tener una valoración global de los temas planteados.

Para futuros proyectos se recomienda complementar el applet, abarcando más clases y objetos dentro de las mismas, junto con sus atributos y métodos lo cual brindará al estudiante mayor comprensión en la temática de concepto de clases y objetos logrando integrar su proceso de enseñanza / aprendizaje.

BIBLIOGRAFÍA

- [1] ALONSO, Catalina M. GALLEGO, Domingo J. HONEY, Peter. Los Estilos de Aprendizaje. Este libro se toman las ideas básicas para comprender y analizar los estilos de aprendizaje
- [2] BLOOM, Benjamín. Taxonomía de los Objetivos de la Educación: Clasificación de las metas Educativas. Manuales I y II. 7 ed. Buenos Aires: El Ateneo, 1979. Este libro describe los niveles de la taxonomía de BLOOM, y plantea los verbos más adecuados para describir los saberes.
- [3] DISEÑO INSTRUCCIONAL Y TEORÍA DEL APRENDIZAJE, Brenda Mergel- Universidad Saskatchewan- Canadá, mayo 1998-Archivo digital en formato PDF. En Windows NT, Active Server Pages, Editorial Ra-Ma, Diciembre 1998.
Conceptos y Orientaciones metodológicas de una propuesta de diseño curricular basado en competencias.
- [4] ESTRADA DIAZ, Lilia Yarley. Elaboración y documentación de una propuesta de diseño curricular bajo la visión de competencias para la asignatura mediciones eléctricas y estudio de su implementación en una plataforma e-learning. Bucaramanga 2005. Trabajo de grado (Ingeniera Electrónica) Universidad Industrial de Santander.
En este libro se presenta el Diseño y elaboración de la estructura curricular para la asignatura mediciones eléctricas bajo una visión de competencias y estudio de adaptación a una plataforma e-learning.
- [5] HERRERA CASTILLO, Jorge. Apuntes de la Programación orientada objetos. 2007.
Define los conceptos, características, funciones y estructuras necesarias para abordar la asignatura programación orientada a objetos.
- [6] L. P. SANTACRUZ-VALENCIA, I. Aedo, C. Delgado Kloos., *a Framework for the Creation, Integration and Reuse of Learning Objects*. IEEE Computer Society Learning Technology Task Force (LTTF) newsletter, Vol. 5 Issue 1. 2003.
Define los diferentes estilos de aprendizaje, basado en tecnologías Learning.
- [7] Peña, C.I., Marzo, J. L., De la Rosa, J. Ll., Fabregat, R. Un sistema de tutoría inteligente adaptativo considerando estilos de aprendizaje, IV

Congreso iberoamericano de informática educativa, IE2002, Vigo (España), Noviembre 20-22, 2002, ISBN 848158-227.
Fundamentos teóricos sobre estilos y modelos de enseñanza/aprendizaje, Tecnologías de Información y Comunicación (TICs), y su aplicación en ambientes de aprendizaje en línea

- [8] PHIL HANNA, SP: Manual de Referencia - McGraw-Hill.
Cómo programar en java: Introducción al D.O.O con UML y los patrones de diseño JDBC, servlets, JSP- Harvey M. Deitel, Paul J- Pearson education.
- [9] WOODS. P.S. Programación de Macromedia Flash TM MX. MacGraw-Hill,2003.
Este libro permite conocer el entorno de Macromedia flash para las animaciones desarrolladas en el proyecto.
- [10] <http://gavilan.uis.edu.co/~clarenes/investigacion.htm>.
Este enlace ofrece una recopilación de publicaciones y experiencias relacionadas con metodologías y entornos de aprendizaje en línea basados en la gestión del conocimiento.
- [11] http://gavilan.uis.edu.co/~clarenes/docencia/guia_didactica.
Este enlace contiene guías didácticas para el diseño y desarrollo de materiales con diferentes estrategias de enseñanza que facilitarán el aprendizaje en línea.
- [12] <http://www.ncsu.edu/felder-public/RMF.html>
Página de Richard Felder, psicólogo que generó el modelo FLSM (Modelo de estilos de aprendizaje de Felder y Silverman) para el manejo de contenidos teniendo en cuenta estilos de aprendizaje; se cuenta con un test de preguntas acerca del modelo de estilos (activo/reflexivo, sensitivo/intuitivo, visual/verbal, y secuencial/global).
- [13] <http://www.aproa.cl/1116/propertyvalue-5538.html>
Este enlace se encuentra la metodología Aproa y su herramienta para la construcción de Objetos de Aprendizaje.
- [14] <http://www.adlnet.gov/index.cfm>
Página Web, que contiene documentación sobre el estándar e-learning SCORM.
- [15] <http://www.frbb.utn.edu.ar/utec/22/n8.html>
Este enlace presenta información relacionada con la formación basada en competencias, y una base para el proyecto estratégico de reforma curricular de las ingenierías.

- [16] <http://tecnologias.gio.etsit.upm.es/elearning>.
Página principal donde se encuentra todo lo relacionado con e- learning.
- [17] <http://fgsnet.nova.edu/cread2/pdf/Alvarado.pdf>
Este documento describe la metodología para los objetos de aprendizaje en el contexto de la educación virtual.
- [18] <http://www.clikear.com/manuales/uml/>
Aquí se puede encontrar documentación para el Desarrollo Orientado a Objetos con UML, lenguaje que permite modelar, construir y documentar los elementos que forman un sistema software orientado a objetos.
- [19] <http://es.tldp.org/Tutoriales/doc-modelado-sistemas-UML/multiple-html/>
Esta guía introduce el Lenguaje Unificado de Modelado (UML), analiza los diagramas que componen UML y ofrece acercamientos a casos de uso guiados sobre cómo estos diagramas se usan para modelar sistemas.
- [20] <http://www.programacion.com/tutorial/csharp/>
Esta obra describe El lenguaje C# el cual es una de las bases de la plataforma Microsoft.NET, no solo lo que corresponde a su sintaxis sino que también se intenta explicar cuáles son las razones que justifican las decisiones tomadas en su diseño y cuáles son los errores más difíciles de detectar que pueden producirse al desarrollar de aplicaciones con él.

ANEXOS

ANEXO A. Cuestionario de estilos de aprendizaje según el modelo de Felder Y Silverman

El ILS de Felder y Silverman está diseñado a partir de cuatro escalas bipolares relacionadas con las preferencias para los estilos de aprendizaje, que en el ILS son *Activo-Reflexivo*, *Sensorial-Intuitivo*, *Visual-Verbal* y *Secuencial-Global*.

Con base en estas escalas, Felder ha descrito la relación de los estilos de aprendizaje con las preferencias de los estudiantes vinculando los elementos de motivación en el rendimiento escolar. El instrumento consta de 44 Ítems y ha sido utilizado, entre otros lugares, en la Universidad del Rosario – Facultad de Educación Continuada en Colombia, en los cursos de educación virtual a partir del año 2001.²²

Encierre en un círculo la opción "a" o "b" para indicar su respuesta a cada pregunta. Por favor seleccione solamente una respuesta para cada pregunta.

Si tanto "a" y "b" parecen aplicarse a usted, seleccione aquella que se aplique más frecuentemente.

1. Entiendo mejor algo

- a) Si lo practico
- b) Si pienso en ello

2. Me considero

- a) realista.
- b) innovador.

3. Cuando pienso acerca de lo que hice ayer, es más probable que lo haga sobre la base de

- a) una imagen.
- b) palabras.

4. Tengo tendencia a

- a) entender los detalles de un tema pero no ver claramente su estructura completa.
- b) entender la estructura completa pero no ver claramente los detalles.

²² Perea Robayo M (2003), Material de estudio para el Diplomado Virtual en Estilos de Aprendizaje de la Universidad del Rosario (Colombia).

5. Cuando estoy aprendiendo algo nuevo, me ayuda
- hablar de ello.
 - pensar en ello.
6. Si yo fuera profesor, yo preferiría dar un curso
- que trate sobre hechos y situaciones reales de la vida.
 - que trate con ideas y teorías.
7. Prefiero obtener información nueva de
- imágenes, diagramas, gráficas o mapas.
 - instrucciones escritas o información verbal.
8. Una vez que entiendo
- todas las partes, entiendo el total.
 - el total de algo, entiendo como encajan sus partes.
9. En un grupo de estudio que trabaja con un material difícil, es más probable que
- participe y contribuya con ideas.
 - no participe y solo escuche.
10. Es más fácil para mí a) aprender hechos.
b) aprender conceptos.
11. En un libro con muchas imágenes y gráficas es más probable que
- revise cuidadosamente las imágenes y las gráficas.
 - me concentre en el texto escrito.
12. Cuando resuelvo problemas de matemáticas
- generalmente trabajo sobre las soluciones con un paso a la vez.
 - frecuentemente sé cuáles son las soluciones, pero luego tengo dificultad para imaginarme los pasos para llegar a ellas.
13. En las clases a las que he asistido
- he llegado a saber cómo son muchos de los estudiantes.
 - raramente he llegado a saber cómo son muchos estudiantes.
14. Cuando leo temas que no son de ficción, prefiero
- algo que me enseñe nuevos hechos o me diga cómo hacer algo.
 - algo que me de nuevas ideas en que pensar.
15. Me gustan los maestros
- que utilizan muchos esquemas en el pizarrón.
 - que toman mucho tiempo para explicar.

16. Cuando estoy analizando un cuento o una novela
- a) pienso en los incidentes y trato de acomodarlos para configurar los temas.
 - b) me doy cuenta de cuáles son los temas cuando termino de leer y luego tengo que regresar y encontrar los incidentes que los demuestran.
17. Cuando comienzo a resolver un problema de tarea, es más probable que
- a) comience a trabajar en su solución inmediatamente.
 - b) primero trate de entender completamente el problema.
18. Prefiero la idea de
- a) certeza.
 - b) Teoría
19. Recuerdo mejor
- a) lo que veo.
 - b) lo que oigo.
20. Es más importante para mí que un profesor
- a) exponga el material en pasos secuenciales claros.
 - b) me dé un panorama general y relacione el material con otros temas.
21. Prefiero estudiar
- a) en un grupo de estudio.
 - b) solo.
22. Me considero
- a) cuidadoso en los detalles de mi trabajo.
 - b) creativo en la forma en la que hago mi trabajo.
23. Cuando alguien me da direcciones de nuevos lugares, prefiero
- a) un mapa.
 - b) instrucciones escritas.
24. Aprendo
- a) a un paso constante. Si estudio con ahínco consigo lo que deseo.
 - b) en inicios y pausas. Me llevo a confundir y súbitamente lo entiendo.
25. Prefiero primero
- a) hacer algo y ver qué sucede.
 - b) pensar cómo voy a hacer algo.
26. Cuando leo por diversión, me gustan los escritores que
- a) dicen claramente los que desean dar a entender.
 - b) dicen las cosas en forma creativa e interesante.

27. Cuando veo un esquema o bosquejo en clase, es más probable que recuerde
- a) la imagen.
 - b) lo que el profesor dijo acerca de ella
28. Cuando me enfrento a un cuerpo de información
- a) me concentro en los detalles y pierdo de vista el total de la misma.
 - b) trato de entender el todo antes de ir a los detalles.
29. Recuerdo más fácilmente
- a) algo que he hecho.
 - b) algo en lo que he pensado mucho.
30. Cuando tengo que hacer un trabajo, prefiero
- a) dominar una forma de hacerlo.
 - b) intentar nuevas formas de hacerlo
31. Cuando alguien me enseña datos, prefiero
- a) gráficas.
 - b) resúmenes con texto.
32. Cuando escribo un trabajo, es más probable que
- a) lo haga (piense o escriba) desde el principio y avance.
 - b) lo haga (piense o escriba) en diferentes partes y luego las ordene.
33. Cuando tengo que trabajar en un proyecto de grupo, primero quiero
- a) realizar una "tormenta de ideas" donde cada uno contribuye con ideas.
 - b) realizar la "tormenta de ideas" en forma personal y luego juntarme con el grupo para comparar las ideas.
34. Considero que es mejor elogio llamar a alguien
- a) sensible.
 - b) imaginativo.
35. Cuando conozco gente en una fiesta, es más probable que recuerde
- a) cómo es su apariencia.
 - b) lo que dicen de sí mismos.
36. Cuando estoy aprendiendo un tema, prefiero
- a) mantenerme concentrado en ese tema, aprendiendo lo más que pueda de él.
 - b) hacer conexiones entre ese tema y temas relacionados.
37. Me considero
- a) abierto.
 - b) reservado.

38. Prefiero cursos que dan más importancia a
a) material concreto (hechos, datos).
b) material abstracto (conceptos, teorías).
39. Para divertirme, prefiero a) ver televisión.
b) leer un libro.
40. Algunos profesores inician sus clases haciendo un bosquejo de lo que enseñarán. Esos bosquejos son
a) algo útil para mí.
b) muy útiles para mí.
41. La idea de hacer una tarea en grupo con una sola calificación para todos
a) me parece bien.
b) no me parece bien.
42. Cuando hago grandes cálculos
a) tiendo a repetir todos mis pasos y revisar cuidadosamente mi trabajo.
b) me cansa hacer su revisión y tengo que esforzarme para hacerlo.
43. Tiendo a recordar lugares en los que he estado a) fácilmente y con bastante exactitud.
b) con dificultad y sin mucho detalle.
44. Cuando resuelvo problemas en grupo, es más probable que yo
a) piense en los pasos para la solución de los problemas.
b) piense en las posibles consecuencias o aplicaciones de la solución en un amplio rango de campos.

INSTRUCCIONES GENERALES PARA CALIFICAR EL INVENTARIO DE ESTILOS DE APRENDIZAJE DE FELDER

- 1) Tome el **Inventario** anterior y una **Hoja de Perfil Individual** en blanco. En la **Hoja de Calificación** asigne UN PUNTO en la casilla correspondiente de acuerdo con el número de la pregunta y su respuesta. Por ejemplo: si su respuesta en la pregunta 5 fue A, coloque 1 en casilla debajo de la letra A y al lado derecho de la pregunta 5.
- 2) Registre de esta manera cada una de las preguntas desde la 1 hasta las 44.
- 3) Luego, sume cada columna y escriba el resultado en la casilla TOTAL COLUMNA.

- 4) Mirando los totales de cada columna por categoría, reste el número menor al mayor.
- 5) Asigne a este resultado la letra en la que obtuvo mayor puntaje en cada categoría.
- 6) Ahora, llene la **Hoja de perfil** con estos resultados, teniendo en cuenta que la letra A corresponde al estilo situado a la izquierda y la letra B al estilo situado a la derecha.
- 7) Finalmente, la Hoja de interpretación permite interpretar los resultados obtenidos.

Si su puntaje en la escala está entre 1 - 3 usted presenta un equilibrio apropiado entre los dos extremos de esa escala.

Si su puntaje está entre 5 - 7 usted presenta preferencia moderada hacia una de los dos extremos de la escala y aprenderá más fácilmente si se le brindan apoyos en esa dirección, (más reflexivo que activo y más global que secuencial).

Si su puntaje en la escala es de 9 - 11 usted presenta una preferencia muy fuerte por uno de los dos extremos de la escala. Usted puede llegar a presentar dificultades para aprender en un ambiente en el cual no cuente con apoyo en esa dirección, (es más intuitivo que sensitivo).

ANEXO B: Contenidos temáticos generales

UNIVERSIDAD INDUSTRIAL DE SANTANDER
FACULTAD DE INGENIERÍAS FÍSICO MECÁNICAS
Escuela de Ingeniería de Sistemas e Informática
Programa de Ingeniería Sistemas

NOMBRE DE LA ASIGNATURA: PROGRAMACIÓN ORIENTADA A OBJETOS

CÓDIGO: 22951

SEMESTRE: II

REQUISITOS: Fundamentos de programación

INTENSIDAD HORARIA SEMANAL: 12

DISCRIMINACIÓN DEL TAD

TADS: Trabajo con acompañamiento docente en el salón de clases: 2

TADL: Trabajo con acompañamiento docente en el Laboratorio: 3

PROPÓSITOS DEL CURSO

- Perfilar la filosofía del diseño orientado por objetos y los conceptos de clase, objeto, herencia, polimorfismo, complejidad, abstracción, encapsulado y ocultamiento.
- Diseñar, codificar, probar y poner a punto programas sencillos de programación orientada a objetos.
- Seleccionar y aplicar modelos de diseño apropiados para construir aplicaciones orientadas a objetos.
- Diseñar y programar como clases contenedoras las estructuras de datos pilas, listas y árboles.
- Desarrollar aplicaciones para ambientes de sistemas de interfaces visuales

JUSTIFICACIÓN

El continuo aprendizaje se fortalece en la investigación, es ahí donde el estudiante como parte de su compromiso con la academia debe enfatizar. Lograr la excelencia profesional sólo es posible si se crea una cultura mental desde los inicios de la carrera, parte de este logro dependerá de los contenidos que el programa ofrece, sin embargo, no se puede obviar la profundización de

la temática que haga el estudiante.

En esa medida, desde el punto de vista de la academia, se considera importante perfilar los contenidos programáticos para que le proporcionen al estudiante herramientas necesarias a fin de convertir las debilidades en fortalezas y así a futuro, encontrar oportunidades de involucrarse con sus diferencias competitivas y comparativas en el ámbito laboral.

CONTENIDO

1. GENERALIDADES

- 1.1 Introducción
- 1.2 Los Sistemas Computacionales
- 1.3 Complejidad
- 1.4 Abstracción
- 1.5 Encapsulado
- 1.6 Reutilización
- 1.7 Calidad del Software
- 1.8 Modulado
- 1.9 Lenguajes de Diseño
- 1.10 Los entornos de programación

2. PILARES DE LA PROGRAMACION ORIENTADA A OBJETOS

- 2.1 Introducción
- 2.2 El Objeto
- 2.3 La Clase
- 2.4 La Herencia
- 2.5 Polimorfismo
- 2.6 Perspectiva histórica de los objetos

3. OBJETOS Y CLASES: DISEÑO Y PROGRAMACION

- 3.1 Introducción
- 3.2 Objeto en el contexto del diseño y programación
- 3.3 Clases en el contexto del diseño y programación
- 3.4 Los Métodos
- 3.5 La visibilidad
- 3.6 El paso de mensajes
- 3.7 Eventos
- 3.8 Diseño y programación de aplicaciones orientado a objetos

4. LA HERENCIA: DISEÑO Y PROGRAMACION

- 4.1 Introducción
- 4.2 Generalización y especialización

- 4.3 Los árboles de la herencia
- 4.4 La Herencia en el contexto del diseño
- 4.5 La Herencia en el contexto de la programación
- 4.6 La Visibilidad de los derechos heredados
- 4.7 Diseño y programación de aplicaciones orientadas a objetos

5. EL POLIMORFISMO: DISEÑO Y PROGRAMACION

- 5.1 Introducción
- 5.2 El Polimorfismo en los métodos
- 5.3 El Polimorfismo en los operadores
- 5.4 Programación de métodos virtuales
- 5.5 Clases Abstractas
- 5.6 Diseño y programación de aplicaciones orientadas a objetos

6. LA PERSISTENCIA DE LOS OBJETOS

- 6.1 Introducción
- 6.2 El modelo de la persistencia con archivos de disco
- 6.3 Programación del modelo de la persistencia con archivos
- 6.4 El modelo de la persistencia con bases de datos
- 6.5 Diseño y programación de aplicaciones orientadas a objetos

7. LAS RELACIONES ENTRE OBJETOS

- 7.1 Introducción
- 7.2 Agregación y composición: Modelado
- 7.3 Asociaciones: Modelado
- 7.4 Multiplicidad: Modelado
- 7.5 Diseño y Programación de aplicaciones orientadas a objetos

ESTRATEGIAS PEDAGOGICAS Y CONTEXTOS POSIBLES DE APRENDIZAJE PARA HORAS TAD Y TI

Este curso se desarrolla mediante exposiciones del profesor con participación de los estudiantes en la discusión de conceptos y solución de problemas y con prácticas en el laboratorio de computadores. Las exposiciones teóricas se realizan en dos horas y las prácticas en tres horas semanales. Entre las estrategias pedagógicas para el logro de los propósitos están:

- Uso de Bibliotecas de clases para que el estudiante analice y utilice su código.
- Laboratorios con prácticas diseñadas y probadas previamente por el grupo de profesores del área.
- Talleres para trabajos en equipo.
- Enunciados de problemas para que el estudiante los solucione.
- El portal del profesor.

EVALUACION DE LA ASIGNATURA

Se realizarán en el semestre mínimo tres evaluaciones. Debe haber por lo menos dos evaluaciones escritas. Podrán ser realizados también: trabajos, quices, laboratorios, informes de laboratorio, exposiciones por parte del alumno. Las ponderaciones para cada una de las evaluaciones serán asignadas por el profesor.

BIBLIOGRAFIA BASICA Y COMPLEMENTARIA

SCHILDT, Herbert. Manual de referencia C#. McGraw-Hill

GUILAR, Luis Joyanes. Programación Orientada a objetos, McGraw-Hill.

MEYER Bertrand. Construcción de Software Orientado a Objetos. Prentice Hall.

CEVALLOS, Francisco Javier. Microsoft C# Lenguajes y aplicaciones. Alfaomega-Rama.

LARMAN, Craig. UML y Patrones. Prentice Hall.

SHARP, John / JAGGER, Jon. Microsoft Visual C#. NET Aprenda YA. McGraw-Hill.

RODRÍGUEZ Miguel / BESTEIRO Marco Antonio. Desarrollo de aplicaciones .NET con Visual C#. McGraw-Hill.

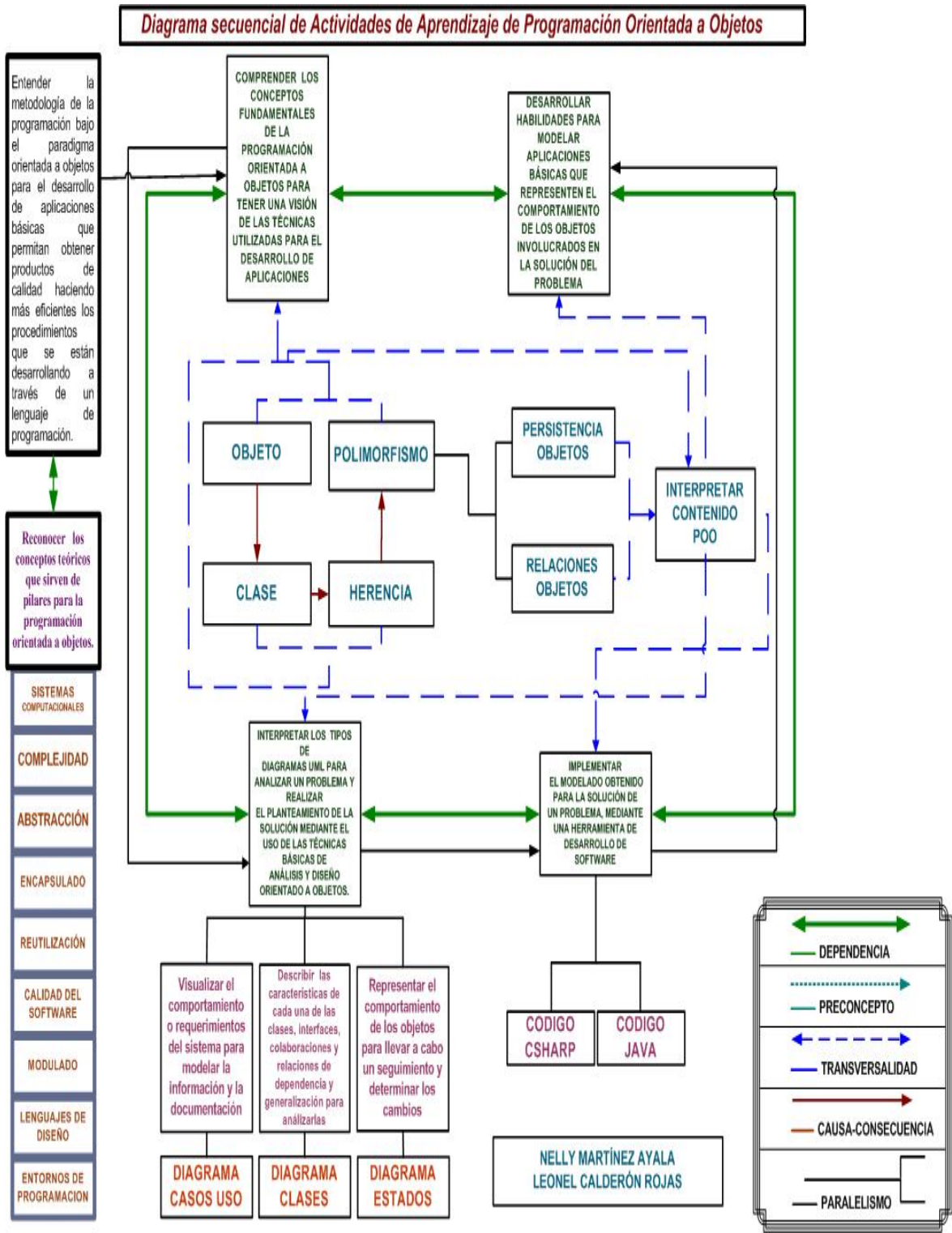
HERRERA CASTILLO, Jorge. Apuntes de la Programación orientada objetos.

RUMBAUGH, James / JACOBSON, Ivar / BOOCH Grady. El Lenguaje Unificado de Modelado. Manual de referencia. Addison Wesley.

RUMBAUGH, James / JACOBSON, Ivar / BOOCH Grady. El Lenguaje Unificado de Modelado. Addison Wesley.

RUMBAUGH, James / JACOBSON, Ivar / BOOCH Grady. El Lenguaje Unificado de Modelado. Introducción. AddisonWesley.

ANEXO C: Diagrama secuencial de actividades de aprendizaje DSA²



ANEXO D: Planteamiento de los saberes y haceres relacionados





 ESCUELA DE INGENIERÍA DE SISTEMAS E INFORMÁTICA   		
Tabla de Saberes Asignatura: Programación Orientada a Objetos	Versión : FINAL	Página 1 de 4
COMPRENDER LOS CONCEPTOS FUNDAMENTALES DE LA PROGRAMACION ORIENTADA A OBJETOS PARA TENER UNA VISIÓN DE LAS TÉCNICAS UTILIZADAS EN EL DESARROLLO DE APLICACIONES		
SABER	HACER	
1. Recapitular conceptos de fundamentos de programación: sistemas computacionales, complejidad, abstracción, encapsulado, reutilización, calidad software, modulado, lenguajes de diseño y entornos de programación.	a. Asociar la información suministrada al entorno de programación orientada a objetos. [1]	
2. Analizar la evolución histórica de la programación orientada a objetos.	b. Comparar el manejo de los datos a través de la evolución de la programación orientada a objetos.[2]	
3. Definir el modelo del paradigma de la programación orientada a objetos.	c. Identificar las características del paradigma de la programación orientada a objetos.[3]	

Tabla de Saberes Asignatura: Programación Orientada a Objetos		Versión : FINAL	Página 2 de 4
4. Establecer los pilares de la programación orientada a objetos.	d. Analizar el funcionamiento de los pilares en el entorno de desarrollo de la programación orientada a objetos. [4]		
5. Indicar las ventajas de la programación orientada a objetos en el desarrollo de software.	e. Describir las herramientas y metodologías para lograr productos competitivos y confiables. [5]		
ANALIZAR UN PROBLEMA Y REALIZAR EL PLANTEAMIENTO DE LA SOLUCIÓN MEDIANTE EL USO DE LAS TÉCNICAS BÁSICAS DE ANÁLISIS Y DISEÑO ORIENTADO A OBJETOS			
SABER		HACER	
6. Describir dentro del contexto del diseño y la programación un Objeto y una Clase.	f. Aplicar el significado de Objeto y Clase en el entorno de los lenguajes de programación. [4] [6]		
7. Definir el concepto y las características de Los Métodos, La Visibilidad, El Paso de Mensajes y Los Eventos.	g. Identificar los propósitos de Los Métodos, La Visibilidad, El Paso de Mensajes y Los Eventos. [4] [7]		
8. Distinguir la Herencia como una propiedad de la programación orientada a objetos.	h. Representar gráficamente el significado de la Herencia. [4] [8]		
9. Interpretar el Polimorfismo dentro de un programa.	i. Especificar los Métodos y las Operaciones con diferentes tipos de datos y objetos. [4] [9]		
	j. Clasificar el polimorfismo dentro de una aplicación. [4] [9]		
10. Entender el concepto de persistencia de un objeto.	k. Distinguir la persistencia de los objetos con archivos y bases de datos. [10]		

Tabla de Saberes Asignatura: Programación Orientada a Objetos		Versión : FINAL	Página 3 de 4
11. Comprender las estructuras utilizadas para establecer las relaciones entre los objetos: Agregación, Composición, Asociación y Multiplicidad.	l. Representar gráficamente las estructuras de relaciones entre los objetos. [11]		
12. Analizar los componentes de los diagramas UML.	m. Interpretar cada uno de los diagramas UML de: Casos de Uso, de Clases y Estados. [4] [8] [9] [12]		
IMPLEMENTAR LA SOLUCIÓN DE UN PROBLEMA, MEDIANTE UNA HERRAMIENTA DE DESARROLLO DE SOFTWARE			
SABER		HACER	
13. Comprender el código de los lenguajes de programación C Sharp y Java.	n. Establecer las diferencias entre los lenguajes de programación C Sharp y Java. [1] [4] [6] [8] [9] [10] [11] [13]		
14. Definir un caso de estudio para el desarrollo dentro de la programación orientada a objetos.	o. Identificar los casos de uso, las clases y los estados a utilizar en el sistema. [4] [6] [8] [9] [14]		
	p. Ejecutar las propiedades del problema a través del lenguaje de programación C Sharp o Java. [1] [4] [6] [8] [9] [10] [11] [13] [14]		
	q. Elaborar ventanas de aplicación en modo consola y Windows mediante el lenguaje de programación C Sharp o Java. [1] [4] [6] [8] [9] [10] [11] [13] [14]		

Tabla de Saberes Asignatura: Programación Orientada a Objetos		Versión : FINAL	Página 4 de 4
DESARROLLAR APLICACIONES BÁSICAS QUE REPRESENTEN EL COMPORTAMIENTO DE LOS OBJETOS INVOLUCRADOS EN LA SOLUCIÓN DEL PROBLEMA			
SABER		HACER	
15. Definir técnicas para representar el comportamiento de los objetos dentro de un lenguaje de Programación.		r. Diseñar la sintaxis en las aplicaciones de software con base a las estructuras jerárquicas. [1] [4][6] [8] [9] [10] [11] [13] [14][15]	
		s. Establecer la solución para el caso de estudio planteado. [1] [4] [6] [8] [9] [10] [11] [13] [14]	
16. Distinguir la reutilización como una propiedad de la Programación orientada a objetos.		t. Establecer la reutilización en el desarrollo de los Programas. [1] [4][6] [11] [13] [14] [16]	
17. Establecer los requerimientos en el desarrollo del Software.		u. Estipular los nuevos requerimientos para obtener Software de calidad. [1] [3] [4] [5][6] [17]	
18. Determinar las características generales de la interfaz de usuario.		v. Establecer en el desarrollo del software la interfaz de usuario. [1] [4][6] [8] [9] [10] [11] [13] [14] [18]	

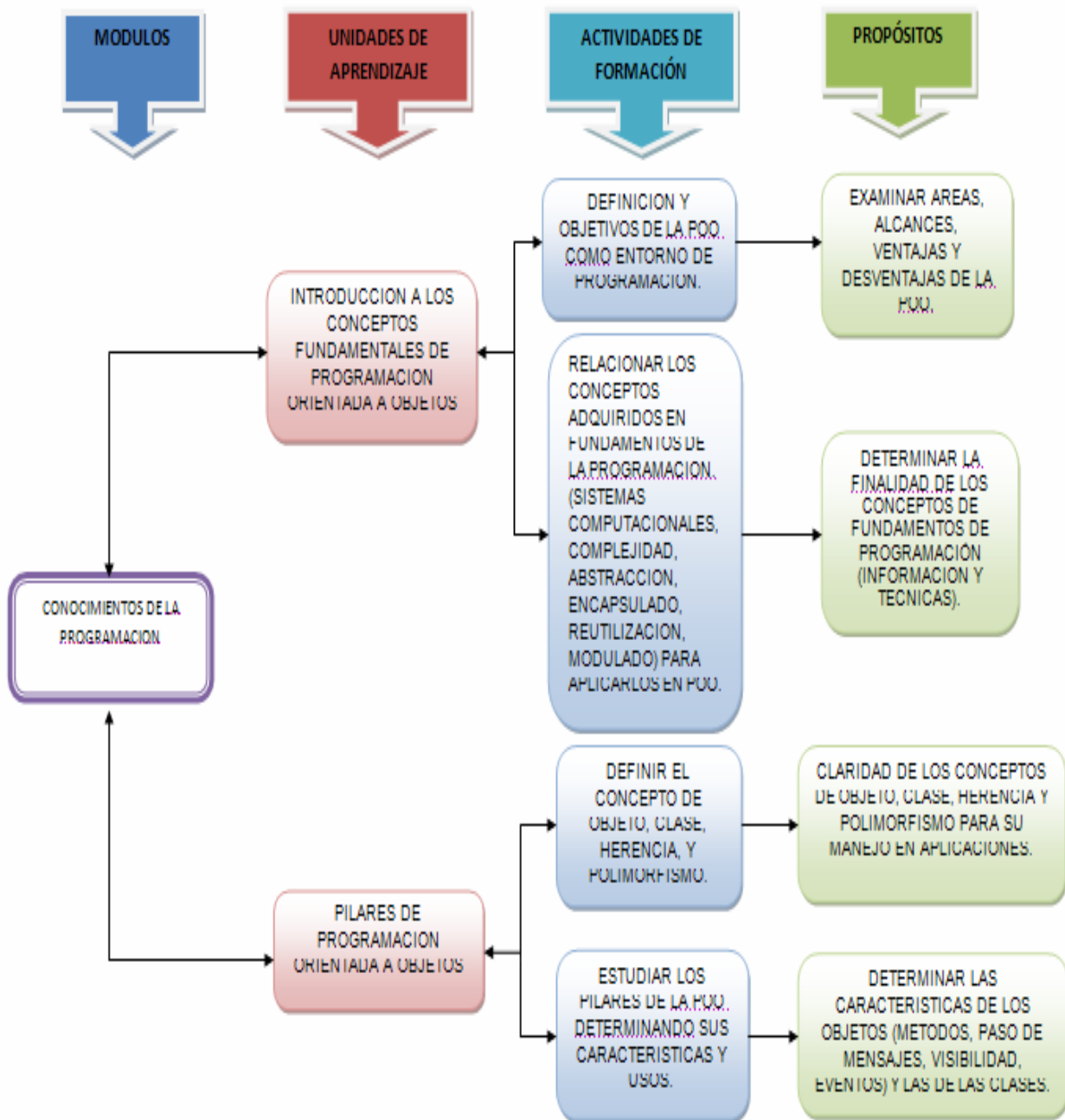
ANEXO E: Establecimiento de la relación Propósitos-Actividades de Formación

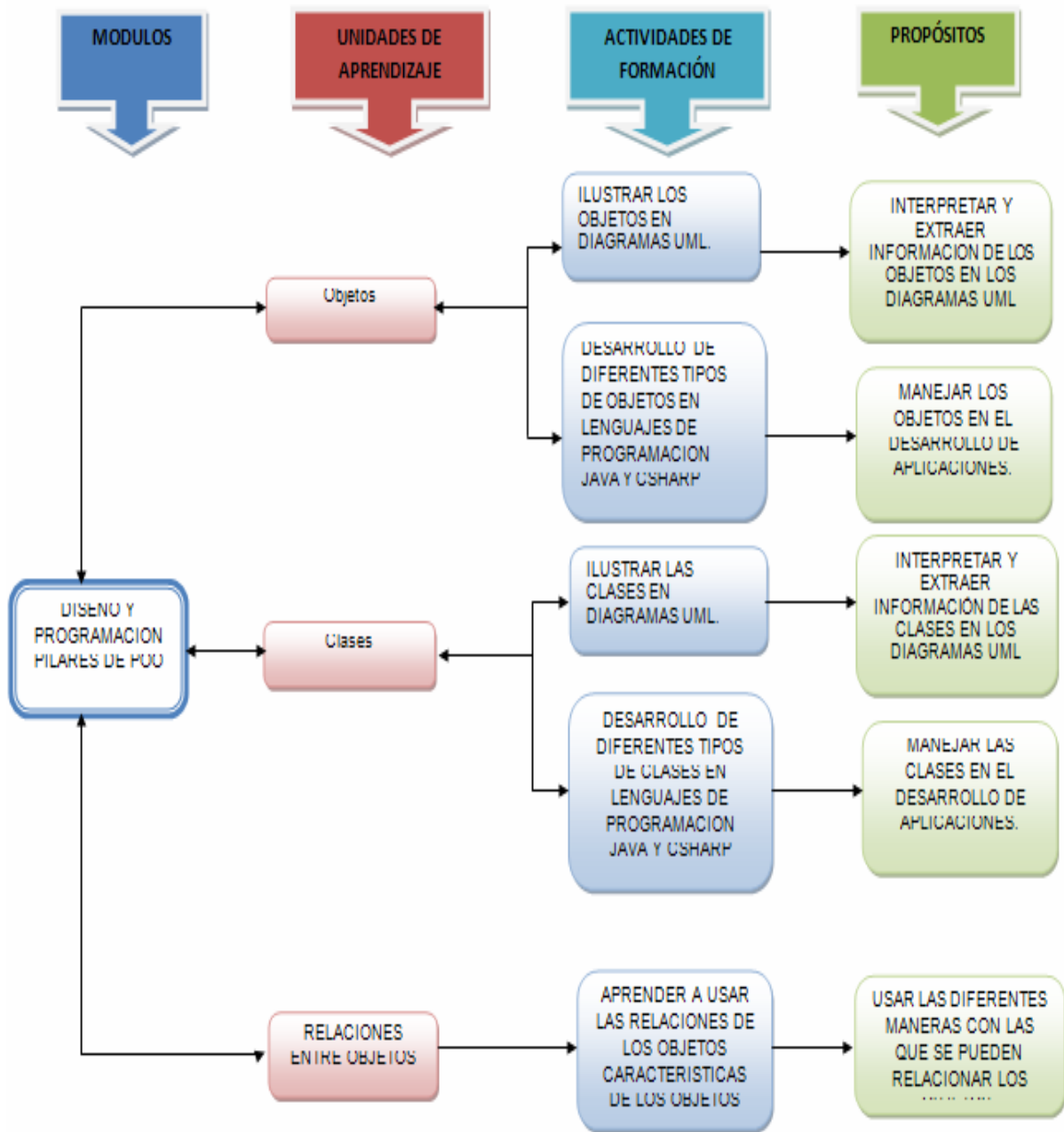
PROPOSITOS – ACTIVIDADES DE FORMACION PROGRAMACIÓN ORIENTADA A OBJETOS		VERSION FINAL	NELLY MARTINEZ LEONEL CALDERON
ACTIVIDADES	PROPOSITOS	SABERES	HACERES
<p>* Definir los objetivos de la POO como entorno de programación.</p> <p>* Relacionar los conceptos adquiridos en fundamentos de la programación. (Sistemas computacionales, complejidad, abstracción, encapsulado, reutilización, modulado) para aplicarlos en POO.</p>	<p>* Examinar áreas, alcances, ventajas y desventajas de la POO.</p> <p>* Determinar la finalidad de los conceptos de fundamentos de programación (información y técnicas).</p>	<p>1. Recapitular conceptos de fundamentos de programación: sistemas computacionales, complejidad, abstracción, encapsulado, reutilización, calidad software, modulado, lenguajes de diseño y entornos de programación.</p> <p>2. Analizar la evolución histórica de la programación orientada a objetos.</p>	<p>a. Asociar la información suministrada al entorno de programación orientada a objetos. [1]</p> <p>b. Comparar el manejo de los datos a través de la evolución de la programación orientada a objetos. [2]</p>
<p>* Definir el concepto de objeto, clase, herencia, y polimorfismo.</p> <p>* Estudiar los pilares de la POO determinando sus características y usos.</p>	<p>* Claridad de los conceptos de objeto, clase, herencia y polimorfismo para su manejo en aplicaciones.</p> <p>* Determinar las características de los objetos (métodos, paso de mensajes, visibilidad, eventos) y las de las clases.</p>	<p>3. Definir el modelo del paradigma de la programación orientada a objetos.</p> <p>4. Establecer los pilares de la programación orientada a objetos.</p> <p>5. Indicar las ventajas de la programación orientada a objetos en el desarrollo de software.</p> <p>6. Describir dentro del contexto del diseño y la programación un Objeto y una Clase.</p> <p>7. Definir el concepto y las</p>	<p>c. Identificar las características del paradigma de la programación orientada a objetos. [3]</p> <p>d. Analizar el funcionamiento de los pilares en el entorno de desarrollo de la programación orientada a objetos. [4]</p> <p>e. Describir las herramientas y metodologías para lograr productos competitivos y confiables. [5]</p>

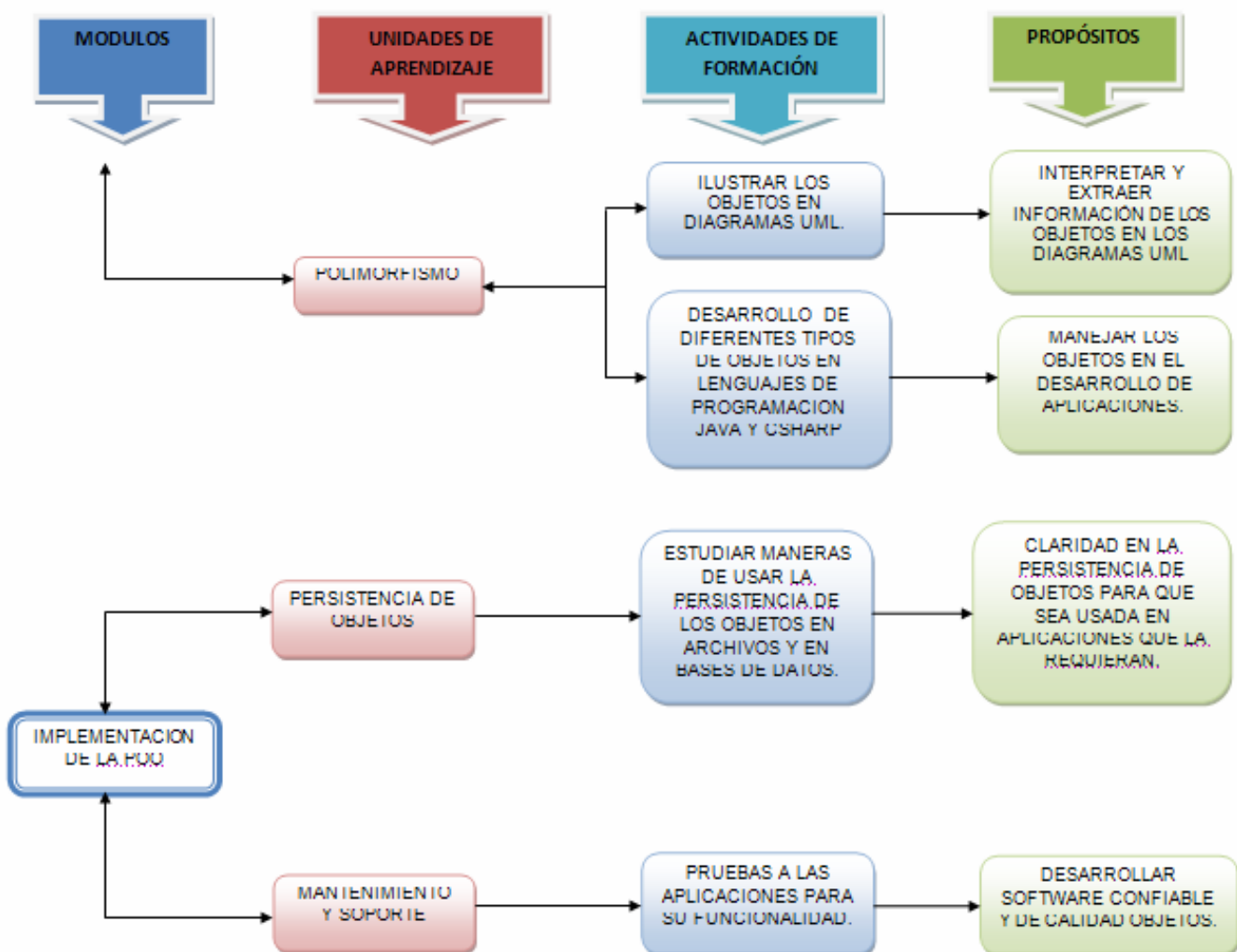
		<p>características de Los Métodos, La Visibilidad, El Paso de Mensajes y Los Eventos.</p> <p>8. Distinguir la Herencia como una propiedad de la programación orientada a objetos.</p>	<p>f. Aplicar el significado de Objeto y Clase en el entorno de los lenguajes de programación. [4] [6]</p> <p>g. Identificar los propósitos de Los Métodos, La Visibilidad, El Paso de Mensajes y Los Eventos. [4] [7]</p> <p>h. Representar gráficamente el significado de la Herencia. [4] [8]</p> <p>i. Especificar los Métodos y las Operaciones con diferentes tipos de datos y objetos. [4] [9]</p>
<p>* Ilustrar los objetos en diagramas uml.</p> <p>* Desarrollar diferentes tipos de objetos en lenguajes de programación Java y Csharp</p>	<p>* Interpretar y extraer información de los objetos en los diagramas uml.</p> <p>* Manejar los objetos en el desarrollo de aplicaciones.</p>	<p>9. Interpretar el Polimorfismo dentro de un programa.</p> <p>10. Entender el concepto de persistencia de un objeto.</p>	<p>j. Distinguir la persistencia de los objetos con archivos y bases de datos. [10]</p> <p>k. Distinguir la persistencia de los objetos con archivos y bases de datos. [10]</p>
<p>* Ilustrar clases en diagramas uml.</p> <p>* Desarrollar de diferentes tipos de clases en lenguajes de programación</p>	<p>* Interpretar y extraer información de las clases en los diagramas uml</p> <p>* Manejo las clases en desarrollo de aplicaciones.</p>	<p>11. Comprender las estructuras utilizadas para establecer las relaciones entre los objetos: Agregación, Composición, Asociación y Multiplicidad.</p>	<p>l. Representar gráficamente las estructuras de relaciones entre los objetos. [11]</p>
<p>* Ilustrar la herencia en diagramas uml.</p> <p>* Desarrollar de la herencia en lenguajes de programación</p>	<p>* Interpretar y extraer información de la herencia en los diagramas uml</p> <p>* Manejo de la herencia para su uso en el desarrollo de aplicaciones.</p>	<p>12. Analizar los componentes de los diagramas UML.</p> <p>13. Comprender el código de los lenguajes de programación C Sharp y Java.</p>	<p>m. Interpretar cada uno de los diagramas UML de: Casos de Uso, de Clases y Estados. [4] [8] [9] [12]</p> <p>n. Establecer las diferencias entre los lenguajes de programación C Sharp y Java. [1] [4] [6] [8] [9] [10]</p>

			[11][13]
* Ilustrar el polimorfismo en diagramas uml. * Desarrollar polimorfismo en lenguajes de programación	* Interpretar y extraer información del polimorfismo en los diagramas uml * Manejo en el concepto de polimorfismo para el uso en el desarrollo de aplicaciones.	14. Definir un caso de estudio para el desarrollo dentro de la programación orientada a objetos.	o. Identificar los casos de uso, las clases y los estados a utilizar en el sistema. [4] [6] [8] [9] [14] p. Ejecutar las propiedades del problema a través del lenguaje de programación C Sharp o Java. [1] [4] [6] [8] [9] [10] [11] [13] [14] q. Elaborar ventanas de aplicación en modo consola y Windows mediante el lenguaje de programación C Sharp o Java. [1] [4] [6] [8] [9] [10] [11] [13] [14]
Estudiar maneras de usar la persistencia de los objetos en archivos y en bases de datos.	Claridad en la persistencia de objetos para que sea usada en aplicaciones que la requieran.	15. Definir técnicas para representar el comportamiento de los objetos dentro de un lenguaje de Programación.	r. Diseñar la sintaxis en las aplicaciones de software con base a las estructuras jerárquicas. [1] [4] [6] [8] [9] [10] [11] [13] [14] [15] s. Establecer la solución para el caso de estudio planteado. [1] [4] [6] [8] [9] [10] [11] [13] [14]
Aprender a usar las relaciones de los objetos características de los objetos	Usar las diferentes maneras con las que se pueden relacionar los objetos.	16. Distinguir la reutilización como una propiedad de la Programación orientada a objetos.	t. Establecer la reutilización en el desarrollo de los Programas. [1] [4] [6] [11] [13] [14] [16]
Probar las aplicaciones para su funcionalidad.	Desarrollar software confiable y de calidad	17. Establecer los requerimientos en el desarrollo del Software	u. Estipular los nuevos requerimientos para
		18. Determinar las características generales de la interfaz de usuario.	obtener Software de calidad. [1] [3] [4] [5] [6] [17] v. Establecer en el desarrollo del software la interfaz de usuario. [1] [4] [6] [8] [9] [10] [11] [13] [14] [18]

ANEXO F: Estructuración Modular







ANEXO F: Planeación Curricular



PLANEACIÓN CURRICULAR
PROGRAMACIÓN ORIENTADA A
OBJETOS



Modulo de formación	Conocimientos de la programación.
Unidad de aprendizaje	Identificar los conceptos fundamentales de la programación orientada a objetos (POO)

Actividad de enseñanza-aprendizaje	Analizar e interpretar las nociones de la programación orientada a objetos.		
Escenarios	Aula de clases, salas informática, salones de conferencia o auditorios.	Duración	12 días
Propósito	Metodología de enseñanza-aprendizaje		
	Estrategia de enseñanza aprendizaje	Métodos	
Examinar áreas, alcances, ventajas y desventajas de la programación orientada a objetos y determinar la finalidad de los conceptos de fundamentos de programación (información y técnicas).	<ol style="list-style-type: none"> 1. Aprendizaje interactivo. 2. Aprendizaje individual. 3. Aprendizaje colaborativo. 4. Aprendizaje por descubrimiento. 5. Aprendizaje basado en problemas 	<ol style="list-style-type: none"> A. Conferencia con un experto(1) B. Consulta(2,3) C. Investigaciones(3,4) D. Análisis y resolución de problemas (3,5) E. Tutorial(3) F. Tareas individuales(2) 	



Evidencias	Estrategias de evaluación	
De conocimiento	Técnicas	Instrumentos
Definición de la programación orientada a objetos	<ol style="list-style-type: none"> 1. Entrevista 2. Prueba o examen. 3. Exposición 	<ol style="list-style-type: none"> A. Cuestionario formal(1) B. Cuestionario(1,2) C. Informe(3)
Objetivos de la programación orientada a objetos	<ol style="list-style-type: none"> 1. Entrevista 2. Prueba o examen. 3. Exposición 	<ol style="list-style-type: none"> A. Cuestionario formal(1) B. Cuestionario(1,2) C. Informe(3)
Ventajas y desventajas de la POO como herramienta de programación	<ol style="list-style-type: none"> 1. Entrevista 2. Prueba o examen. 3. Exposición 	<ol style="list-style-type: none"> A. Cuestionario formal(1) B. Cuestionario(1,2) C. Informe(3)
Conceptos en sistemas computacionales, complejidad, abstracción, encapsulado, reutilización, modulado para su manejo en aplicaciones.	<ol style="list-style-type: none"> 1. Ensayo 2. Prueba o examen 3. Actividades complementarias 4. Practica de laboratorio 	<ol style="list-style-type: none"> A. Ensayo(1) B. Ejercicios(2,3) C. Algoritmo(4, 5) D. Taller de problemas(2, 3)

	5. Diagramas de información.	
De desempeño	Técnicas	Instrumentos
Ubica la programación orientada a objetos entre los lenguajes más confiables y completos.	1. Entrevista 2. Prueba o examen. 3. Exposición	A. Cuestionario formal(1) B. Cuestionario(1,2) C. Informe(3)
Información y técnicas de fundamentos de programación que sirven de habilidades en el desarrollo de aplicaciones POO	1. Prueba o examen	A. Cuestionario(1) B. Ejercicios(1)
De producto	Técnicas	Instrumentos
Aplica información y técnicas de programación estructurada y por eventos	1. Prueba o examen 2. Practica de laboratorio	A. Ejercicios(1) B. Taller de problemas(1) C. Cuestionario(1) D. Algoritmo(2)



Diseño de los medios didácticos para el objeto de aprendizaje	
Núcleo del conocimiento	Documentos como introducción la P.O.O, el paradigma de la P.O.O, composición de una aplicación, construcción de una aplicación.
Pdf	Construcción de programas. Conceptos generales. Fundamentos de la programación
Video	Alcances de la programación orientada a objetos Metas de la programación orientada a objetos.
Audio	Construcción de programas. Conceptos generales. Fundamentos de la programación.
Grafico	Diagramas flujo, diagramas que muestren el uso de los conceptos
Aplicativo	Aplicación que interactúe con las características de los conceptos de fundamentos de la programación y que ilustre un posible resultado y su relación con la POO

Modulo de formación	Conocimientos de la programación
Unidad de aprendizaje	Pilares de la programación orientada a objetos

Actividad de enseñanza-aprendizaje	Identificar los pilares de la programación orientada a objetos		
Escenarios	Aula de clases, salas informática, salones de conferencia o auditorios.	Duración	12 días
Propósito	Metodología de enseñanza-aprendizaje		
	Estrategia de enseñanza aprendizaje		Métodos
Claridad en los conceptos de objeto, clase, herencia y polimorfismo para su manejo en aplicaciones.	<ol style="list-style-type: none"> 1. Aprendizaje interactivo 2. Aprendizaje colaborativo 3. Aprendizaje individual 4. Aprendizaje por descubrimiento 5. Aprendizaje significativo 6. Aprendizaje basado en problemas 	<ol style="list-style-type: none"> A. Conferencia con un experto(1) B. Análisis e interpretación de lectura(2, 3) C. Consulta(2, 3) D. Investigaciones(2, 4) E. Ilustraciones(5) F. Diagramas(5) G. Solución de casos(6) 	

Evidencias	Estrategias de evaluación	
	Técnicas	Instrumentos
De conocimiento		
Comprende los conceptos y características de los pilares: objeto, clase herencia, polimorfismo para aplicarlos en la programación orientada a objetos.	<ol style="list-style-type: none"> 1. Prueba o examen 2. Actividades complementarias 3. Exposición 	<ol style="list-style-type: none"> A. Cuestionario(1) B. Ejercicios(1, 3) C. Informe(3) D. Resumen(2, 3)
De desempeño		
Analiza conceptos pilares de la POO: objeto, clase herencia, polimorfismo.	<ol style="list-style-type: none"> 1. Prueba o examen 2. Actividades complementarias 3. Exposición 	<ol style="list-style-type: none"> A. Cuestionario(1) B. Ejercicios(1, 3) C. Informe(3) D. Resumen(2, 3)
De producto		
Sólida base de conceptos y características que luego servirán en el diseño y	<ol style="list-style-type: none"> 1. Prueba o examen 2. Actividades complementarias 3. Exposición 	<ol style="list-style-type: none"> A. Cuestionario(1) B. Ejercicios(1, 3) C. Informe(3)

programación		D. Resumen(2, 3)
--------------	--	------------------

Diseño de los medios didácticos para el objeto de aprendizaje	
Núcleo del conocimiento	Documentación de los pilares de la programación orientada a objetos acerca de sus usos y características
Pdf	Pilares de la programación orientada a objetos(objetos y clases) Herencia Polimorfismo
Video	Video tutorial de definiciones y principios de código de objeto, clase, herencia y polimorfismo
Audio	Definiciones de los pilares de la programación orientada a objetos
Grafico	Diagramas UML objetos, clases, herencia, polimorfismo jerarquía
Aplicativo	Applet de clases y objetos que interactúe con los atributos y métodos de cada uno de este modo se genere el código java de la clase y el objeto con su respectivo diseño UML y muestre su resultado grafico.

+

Modulo de formación	Diseño y programación pilares de POO
Unidad de aprendizaje	Objetos, clases, herencia, polimorfismo

Actividad de enseñanza-aprendizaje	Comprender los diferentes diagramas UML y sus usos		
Escenarios	Aula de clases, salas informática, salones de conferencia o auditorios.	Duración	12 días
Propósito	Metodología de enseñanza-aprendizaje		
	Estrategia de enseñanza aprendizaje	Métodos	
Interpretar y extraer información de los objetos, las clases, la herencia y el polimorfismo en los diagramas UML	<ol style="list-style-type: none"> 1. Aprendizaje significativo 2. Aprendizaje basado en problemas 3. Aprendizaje colaborativo 4. Aprendizaje individual 	<ol style="list-style-type: none"> A. Diagramas(1) B. Ilustraciones(2) C. Resolución y análisis de ejercicios(3) D. Análisis e interpretación de lectura(4, 5) C. Tareas individuales(5) 	

Evidencias	Estrategias de evaluación
-------------------	----------------------------------

De conocimiento	Técnicas	Instrumentos
Conceptos y análisis de las estructuras en los diagramas UML	1. Práctica de laboratorio 2. Actividades complementarias 3. Diagramas de información 4. Prueba o examen	A. Algoritmo(1, 3) B. Ejercicios(2, 4) C. Taller de problemas(4)
De desempeño	Técnicas	Instrumentos
Clasifica los diferentes tipo de diagramas UML	1. Práctica de laboratorio 2. Actividades complementarias 3. Prueba o examen	A. Cuestionario(1) B. Ejercicios(2, 3) C. Taller de problemas(3)
De producto	Técnicas	Instrumentos
Interpretación de las estructuras en los diagramas UML para su uso en la toma de requisitos e información para una aplicación.	1. Práctica de laboratorio 2. Actividades complementarias 3. Diagramas de información 4. Prueba o examen	A. Algoritmo(1, 3) B. Ejercicios(2, 4) C. Taller de problemas(4)

Diseño de los medios didácticos para el objeto de aprendizaje	
Núcleo del conocimiento	Generalidades del lenguaje UML, modelado UML.
Pdf	Generalidades del lenguaje UML, modelado UML.
Video	Video tutorial de creación de diagramas UML
Audio	Conceptos de las generalidades del modelado en UML
Grafico	Diagramas UML de ejercicios particulares de todas las áreas
Aplicativo	Aplicación interactiva que muestren los pasos a seguir en la construcción de diagramas básicos UML

Modulo de formación	Diseño y programación pilares de POO
Unidad de aprendizaje	Objetos, clases, herencia, relación de objetos, polimorfismo



Actividad de enseñanza-aprendizaje	Desarrollo de aplicaciones en lenguaje orientado a objetos java o csharp		
Escenarios	Aula de clases, salas informática, salones de conferencia o auditorios.	Duración	30 días
Propósito	Metodología de enseñanza-aprendizaje		

	Estrategia de enseñanza aprendizaje	Métodos
Manejo de los objetos, las clases, la herencia, el polimorfismo y la relación entre objetos en el desarrollo de aplicaciones.	1. Aprendizaje significativo 2. Aprendizaje basado en problemas 3. Aprendizaje colaborativo 4. Aprendizaje individual 5. Aprendizaje por descubrimiento	A. Diagramas(1) B. Ilustraciones(2) C. Resolución y análisis de ejercicios(3) D. Análisis e interpretación de lectura(4, 5) C. Tareas individuales(5) E. Solución de casos(2) F. Practica de laboratorio(5)

Evidencias	Estrategias de evaluación	
	Técnicas	Instrumentos
De conocimiento		
Introducción de técnicas avanzadas del desarrollo orientado a objetos	1. Práctica de laboratorio 2. Actividades complementarias 3. Diagramas de información 4. Prueba o examen	A. Algoritmo(1, 3) B. Ejercicios(2, 4) C. Taller de problemas(4)
Conocimiento y empleo de un lenguaje de programación orientado a objetos (java, visual csharp) con el que plasmar en el terreno práctico los conceptos expuestos en la teoría	1. Práctica de laboratorio 2. Actividades complementarias 3. Diagramas de información 4. Prueba o examen	A. Algoritmo(1, 3) B. Ejercicios(2, 4) C. Taller de problemas(4)
Crear objetos (atributos, métodos) y reconocer la interrelación entre ellos.	1. Práctica de laboratorio 2. Actividades complementarias 3. Diagramas de información 4. Prueba o examen 5. Proyectos	A. Algoritmo(1, 3) B. Ejercicios(2, 4) C. Taller de problemas(4) D. Informe
Declarar clases(constructores, destructores, tipo de métodos)	1. Práctica de laboratorio 2. Actividades complementarias 3. Diagramas de información 4. Prueba o examen 5. Proyectos	A. Algoritmo(1, 3) B. Ejercicios(2, 4) C. Taller de problemas(4) D. Informe
Clases derivadas(herencia simple y múltiple)	1. Práctica de laboratorio 2. Actividades complementarias 3. Diagramas de información	A. Algoritmo(1, 3) B. Ejercicios(2, 4) C. Taller de problemas(4)

	4. Prueba o examen	
Manipular el polimorfismo donde se observe y representarlo en lenguaje de programación	1. Práctica de laboratorio 2. Actividades complementarias 3. Diagramas de información 4. Prueba o examen	A. Algoritmo(1, 3) B. Ejercicios(2, 4) C. Taller de problemas(4)
De desempeño	Técnicas	Instrumentos
Habilidad de programación en los lenguajes orientados a objetos.	1. Práctica de laboratorio 2. Actividades complementarias 3. Diagramas de información 4. Prueba o examen	A. Algoritmo(1, 3) B. Ejercicios(2, 4) C. Taller de problemas(4)
De producto	Técnicas	Instrumentos
Práctica en el desarrollo de sistemas orientados a objetos.	1. Práctica de laboratorio 2. Actividades complementarias 3. Diagramas de información 4. Prueba o examen	A. Algoritmo(1, 3) B. Ejercicios(2, 4) C. Taller de problemas(4)

Diseño de los medios didácticos para el objeto de aprendizaje	
Núcleo del conocimiento	Implementación en código java o visual csharp
Pdf	Java, visual csharp
Video	Video tutoriales o animaciones de procedimientos en código java o csharp
Audio	Líneas de código java o csharp
Grafico	Diagramas flujo, UML de ejemplos particulares que muestren todas las áreas de la POO.
Aplicativo	Aplicación de generación código java o c# simples y básicos para cada concepto de la POO. Paso a paso pedirá las características de un ejemplo particular, e interactuara con, la clase, la herencia, el polimorfismo y la relación de objetos.

Modulo de formación	Implementación de la programación orientada a objetos
Unidad de aprendizaje	Persistencia de objetos

Actividad de enseñanza-aprendizaje		El problema de la persistencia de objetos y su solución.	
Escenarios	Aula de clases, salas informática, salones de conferencia o auditorios	Duración	15 días
Propósito	Metodología de enseñanza-aprendizaje		
	Estrategia de enseñanza aprendizaje		Métodos
Claridad en la persistencia de objetos para que sea usada en aplicaciones que la requieran.	<ol style="list-style-type: none"> 1. Aprendizaje interactivo 2. Aprendizaje individual 3. Aprendizaje colaborativo 4. Aprendizaje por descubrimiento 5. Aprendizaje basado en problemas 	<ol style="list-style-type: none"> A. Conferencia con un experto(1) B. Tareas individuales(2) C. Taller de ejercicios(3) D. Practica de laboratorio(4) E. Análisis de ejercicios(5) F. Solución de casos(5) G. Tutorial(3) 	
Evidencias		Estrategias de evaluación	
De conocimiento		Técnicas	Instrumentos
Determinar el tipo de la persistencia analizando los criterios estructurales, funcionales y de organización.		<ol style="list-style-type: none"> 1. Prueba o examen 2. Actividades complementarias 3. Practica de laboratorio 4. Debate 	<ol style="list-style-type: none"> A. Cuestionario(1) B. Ejercicios(1,2) C. Taller de problemas(1,2) D. Informe(3) E. Resumen(4)
El modelo de la persistencia con base de datos		<ol style="list-style-type: none"> 1. Prueba o examen 2. Actividades complementarias 3. Practica de laboratorio 	<ol style="list-style-type: none"> A. Cuestionario(1) B. Ejercicios(1,2) C. Taller de problemas(1,2) D. Informe(3)
El modelo de la persistencia con archivos		<ol style="list-style-type: none"> 1. Prueba o examen 2. Actividades complementarias 3. Practica de laboratorio 	<ol style="list-style-type: none"> A. Cuestionario(1) B. Ejercicios(1,2) C. Taller de problemas(1,2) D. Informe(3)
De desempeño		Técnicas	Instrumentos
Implementación de mecanismos que permitan a los objetos mantenerse vivos tras la finalización del proceso que les dio vida		<ol style="list-style-type: none"> 1. Prueba o examen 2. Actividades complementarias 3. Practica de laboratorio 4. Proyectos 	<ol style="list-style-type: none"> A. Cuestionario(1) B. Ejercicios(1,2) C. Taller de problemas(1,2) D. Informe(3, 4)

		E. Resumen(2)
De producto	Técnicas	Instrumentos
Crear soluciones óptimas a problemas que requieran el uso de la persistencia	<ol style="list-style-type: none"> 1. Prueba o examen 2. Actividades complementarias 3. Practica de laboratorio 4. Proyectos 	<ol style="list-style-type: none"> A. Cuestionario(1) B. Ejercicios(1,2) C. Taller de problemas(1,2) D. Informa(3, 4) E. Resumen(2)

Diseño de los medios didácticos para el objeto de aprendizaje	
Núcleo del conocimiento	Estudio de documentación de persistencia de objetos con archivos o base de datos y análisis de los datos del problema para planteamiento de la solución del problema.
Pdf	Persistencia de objetos
Video	Video tutorial que muestre el manejo de la persistencia de objetos en archivos y en base de datos en cualquiera de los lenguajes java o sharp
Audio	Conceptos, y usos de los tipos de persistencia de objetos
Grafico	Diagramas en donde se puedan observar y entender de una manera más sencilla los diferentes tipos de persistencia en ejemplo particulares.
Aplicativo	Aplicación interactiva que recoja requerimientos y muestre el comportamiento de cada tipo de persistencia por medio de diagramas.

Modulo de formación	Implementación de la POO
Unidad de aprendizaje	Mantenimiento y soporte



Actividad de enseñanza-aprendizaje	Desarrollo de software fácil de mantener, modificar y reutilizar		
Escenarios	Aula de clases, salas informática, salones de conferencia o auditorios.	Duración	15 días
Propósito	Metodología de enseñanza-aprendizaje		
	Estrategia de enseñanza aprendizaje	Métodos	
Desarrollar software confiable y de calidad	<ol style="list-style-type: none"> 2. Aprendizaje basado en problemas 3. Aprendizaje colaborativo 4. Aprendizaje individual 	<ol style="list-style-type: none"> B. Ilustraciones(2) C. Resolución y análisis de ejercicios(3) D. Análisis e interpretación de lectura(4, 5) 	

		C. Tareas individuales(5)
--	--	---------------------------

Evidencias	Estrategias de evaluación	
	Técnicas	Instrumentos
De conocimiento		
Valorar en qué medida las técnicas orientadas a objetos favorecen la calidad del software, analizando sobre todo como facilitan la reutilización y extensibilidad.	<ol style="list-style-type: none"> 1. Prueba o examen 2. Actividades complementarias 3. Practica de laboratorio 4. Proyectos 	<ol style="list-style-type: none"> A. Cuestionario(1) B. Ejercicios(1,2) C. Taller de problemas(1,2) D. Informe(3, 4) E. Resumen(2)
De desempeño		
Conocer las ventajas de la reutilización de código y qué pasos hay que seguir para elaborar arquitecturas de programación que faciliten al máximo esta reutilización	<ol style="list-style-type: none"> 1. Prueba o examen 2. Actividades complementarias 	<ol style="list-style-type: none"> A. Cuestionario(1) B. Ejercicios(1,2) C. Taller de problemas(1,2) E. Resumen(2)
Controlar situaciones en donde el programa no muestre lo esperado o genere errores	<ol style="list-style-type: none"> 1. Prueba o examen 2. Actividades complementarias 3. Practica de laboratorio 4. Proyectos 	<ol style="list-style-type: none"> A. Cuestionario(1) B. Ejercicios(1,2) C. Taller de problemas(1,2) D. Informe(3, 4) E. Resumen(2)
De producto		
Dominar un lenguaje de programación orientado a objetos y todas las herramientas que conformen su entorno de programación para desarrollos eficientes.	<ol style="list-style-type: none"> 1. Prueba o examen 2. Actividades complementarias 3. Practica de laboratorio 4. Proyectos 	<ol style="list-style-type: none"> A. Cuestionario(1) B. Ejercicios(1,2) D. Informe(3, 4) E. Resumen(2)

Diseño de los medios didácticos para el objeto de aprendizaje	
Núcleo del conocimiento	Material de implementación software en programación orientada a objetos: colecciones, enlaces y asociaciones
Pdf	Tutoriales colecciones, enlaces y asociaciones, mantenimiento y soporte a aplicaciones
Video	Alcances del mantenimiento, modificación y reutilización de software
Audio	Técnicas para mantener, reutilizar y modificar software.

		C. Tareas individuales(5)
--	--	---------------------------

Evidencias	Estrategias de evaluación	
	Técnicas	Instrumentos
De conocimiento		
Valorar en qué medida las técnicas orientadas a objetos favorecen la calidad del software, analizando sobre todo como facilitan la reutilización y extensibilidad.	<ol style="list-style-type: none"> 1. Prueba o examen 2. Actividades complementarias 3. Practica de laboratorio 4. Proyectos 	<ol style="list-style-type: none"> A. Cuestionario(1) B. Ejercicios(1,2) C. Taller de problemas(1,2) D. Informe(3, 4) E. Resumen(2)
De desempeño		
Conocer las ventajas de la reutilización de código y qué pasos hay que seguir para elaborar arquitecturas de programación que faciliten al máximo esta reutilización	<ol style="list-style-type: none"> 1. Prueba o examen 2. Actividades complementarias 	<ol style="list-style-type: none"> A. Cuestionario(1) B. Ejercicios(1,2) C. Taller de problemas(1,2) E. Resumen(2)
Controlar situaciones en donde el programa no muestre lo esperado o genere errores	<ol style="list-style-type: none"> 1. Prueba o examen 2. Actividades complementarias 3. Practica de laboratorio 4. Proyectos 	<ol style="list-style-type: none"> A. Cuestionario(1) B. Ejercicios(1,2) C. Taller de problemas(1,2) D. Informe(3, 4) E. Resumen(2)
De producto		
Dominar un lenguaje de programación orientado a objetos y todas las herramientas que conformen su entorno de programación para desarrollos eficientes.	<ol style="list-style-type: none"> 1. Prueba o examen 2. Actividades complementarias 3. Practica de laboratorio 4. Proyectos 	<ol style="list-style-type: none"> A. Cuestionario(1) B. Ejercicios(1,2) D. Informe(3, 4) E. Resumen(2)

127

Diseño de los medios didácticos para el objeto de aprendizaje	
Núcleo del conocimiento	Material de implementación software en programación orientada a objetos: colecciones, enlaces y asociaciones
Pdf	Tutoriales colecciones, enlaces y asociaciones, mantenimiento y soporte a aplicaciones
Video	Alcances del mantenimiento, modificación y reutilización de software
Audio	Técnicas para mantener, reutilizar y modificar software.
Grafico	Diagramas y dibujos informativos que muestren las diferentes situaciones de mantener, modificar y reutilizar software
Aplicativo	Aplicación interactiva donde se pueda colocar algo de código desarrollado y que se generen las posibles soluciones optimas para el mejoramiento del código.