

# Diseño de una asignatura basado en aprendizaje activo que separa el análisis y diseño de la programación orientada a objetos

## Active learning design of a course which separates the analysis and the design from the object oriented programming

Luis Carlos Díaz Chaparro<sup>1</sup>, MSc, Deicy Alvarado<sup>2</sup>, PhD y Angela Carrillo Ramos<sup>1</sup>, PhD

1. Pontificia Universidad Javeriana, Depto de Ingeniería de Sistemas, Colombia

2. Universidad Distrital Francisco José de Caldas, Ingeniería de Sistemas, Colombia

{luis.diaz, angela.carrillo}@javeriana.edu.co; lalvarado@udistrital.edu.co

Recibido para revisión 10 de septiembre de 2009, aceptado 29 de enero de 2010, versión final 12 de febrero de 2010

**Resumen**—En este artículo se muestra el proceso evolutivo evidenciado en el contenido de la asignatura “Análisis y Diseño Orientado a Objetos” (cuyo acrónimo es *ADOO*) que se imparte en la Pontificia Universidad Javeriana, sede Bogotá, Colombia. Actualmente se intenta llegar a que los estudiantes comprendan los conceptos del Paradigma Orientado a Objetos (*OO*) y a que separen dichos conceptos de la programación, de manera que cuando cursen la asignatura “*Programación OO*”, cuenten con mejores herramientas y bases conceptuales con el fin de obtener mejores resultados en ella. A lo largo de este proceso, se ha planteado el uso de recursos de apoyo al aprendizaje y se han desarrollado algunas herramientas y artefactos que han sido de gran utilidad tanto para profesores como para estudiantes.

**Palabras Clave**—Aprendizaje Activo, Análisis, Diseño y Programación OO, Mapas Mentales, GRACE.

**Abstract**—In this paper, we present the evolutionary process demonstrated in the content of the course named Object Oriented Analysis and Design (whose Spanish acronym is *ADOO*) that is imparted at the Pontificia Universidad Javeriana, Bogotá, Colombia. Nowadays, it is tried to arrive at that the students understand the Object Oriented (*OO*) Paradigm concepts in order to that they separate these concepts of the programming, so that when they take the course “Object Oriented Programming”, they have better tools and conceptual bases with the purpose of obtaining better results in this course. Throughout this process, the use of resources of support to the learning has been considered and some tools and devices have been developed; they have been very useful as much for teachers as for students.

**Keywords**—Active Learning, Object Oriented Analysis, Design and Programming, Mind Maps, GRACE.

### I. INTRODUCCIÓN

Los cambios en los contenidos programáticos de las asignaturas impartidas en las diferentes carreras de las universidades colombianas, en general obedecen a lineamientos y tendencias nacionales o internacionales y están siempre acorde con el perfil del egresado que identifica la institución. En Ingeniería de Sistemas concretamente, estos lineamientos están dados por instituciones tales como *ACM*, *IEEE* y a nivel nacional por *ACOFI* [2], las cuales coinciden en señalar la Ingeniería de software como área fundamental y dentro de ella, el Análisis y el Diseño.

Con base en esto, desde el año 2005, la carrera de Ingeniería de Sistemas de la Pontificia Universidad Javeriana (sede Bogotá, Colombia), ha apostado por un cambio de currículo tendiente a fortalecer el proceso de Análisis y Diseño Orientado a Objetos (*ADOO*), separándolo de la programación correspondiente a su implementación. Es así como, debido a lo novedoso de este cambio, no es fácil encontrar material didáctico desarrollado y completo y es precisamente por este motivo que los profesores encargados de la materia, han desarrollado una serie de herramientas y materiales pedagógicos, cuyo objetivo es facilitar la adquisición de los conceptos de Análisis y Diseño por parte de los estudiantes.

En este documento se presenta entonces, la argumentación acerca de la importancia del Análisis y Diseño dentro de Ingeniería de sistemas, así como un compendio de los diferentes elementos y herramientas desarrollados por los profesores de la materia, con el fin de enriquecer y facilitar el proceso de aprendizaje por parte de los estudiantes.

## II. ADOO EN INGENIERÍA DE SISTEMAS

Los grandes avances en tecnologías de la información han hecho que diversos sectores a nivel mundial, se interesen y planteen estándares para el desarrollo de currículos en esta área. Tal es el caso de *Career Space* [11] quienes en principio plantean que los egresados de las carreras relacionadas con las Tecnologías de la Información y la Comunicación (*TIC*) deben aprender a trabajar en equipo y tener experiencias en proyectos que involucren actividades paralelas.

En este mismo sentido, Andreas Kaiser [11] evidencia la necesidad que tienen los graduados de trabajar en equipos con sistemas muy complejos donde existe gran interacción e interdependencia, lo cual hace indispensable que los estudiantes adquieran “*capacidades sistémicas*” [11], las cuales consisten en analizar, representar y aislar problemas con el fin de resolverlos.

Kaiser [11] también muestra que las “*capacidades sistémicas*” que él mismo menciona, no están de manera explícita en los currículos y al parecer no existen herramientas que ayuden a los estudiantes a adquirir dichas capacidades.

A su vez, Peter Revil [15] afirma que dentro de los conceptos fundamentales en la formación en *TIC* está la capacidad de aplicar lo aprendido en otros contextos, lo que él denomina “*Transferencia*” [15]. Dicho autor argumenta que la enseñanza basada en la experiencia ayuda a los estudiantes a contextualizar y a adquirir capacidades conductuales, que según Revil, ayudarán a los egresados a tener una mayor confianza en el desarrollo de las tareas que se esperan de ellos.

Por otra parte, *ACMe IEEE-CS* [3][4] en su publicación *Computing Curricula*, definen de manera consensuada, diversos términos relacionados con las disciplinas de la computación; sin embargo, para el análisis llevado a cabo en este documento, se tendrán en cuenta las definiciones de: Ingeniería de Sistemas y Computación, Análisis y Modelado de software y Diseño de Software de la siguiente manera:

***Ingeniería de Sistemas y Computación*** [4]: Integra elementos de Ingeniería computacional, ciencias de la computación e ingeniería de software. Se enfoca especialmente en el diseño e implementación de sistemas en tiempo real y especificación formal de sistemas.

***Análisis y Modelado de Software*** [4]: Actividad que intenta modelar los requerimientos y restricciones del cliente para definir el problema real a ser resuelto.

***Diseño de Software*** [4]: Actividad que convierte el modelo de requerimientos en un modelo detallado que representa una solución y que incluye diversas especificaciones.

Según *ACMe IEEE-CS* [3][4], los tópicos de mayor relevancia en Ingeniería de Software son: Análisis de Requerimientos Técnicos, Análisis y Modelado de Software, Diseño de Software, Validación y Verificación de Software.

Todo lo anterior, permite concluir que a nivel mundial, se viene planteando la necesidad de dar una mayor preponderancia al Análisis y Diseño de Software dentro de los currículos de Ingeniería de sistemas, apoyando su enseñanza en el desarrollo de capacidades que permitan a los estudiantes llevar a cabo aplicaciones prácticas.

En cuanto a Colombia, a partir del año 2003 el gobierno nacional creó los *ECAES* [10], exámenes de carácter obligatorio, encaminados a evaluar la calidad de los programas de educación pública superior a nivel de pregrado y asignó la responsabilidad de su implementación al *ICFES* [10] (Instituto Colombiano para el fomento de la educación superior). Debido a esto, *ACOFI* (acrónimo de Asociación Colombiana de Facultades de Ingeniería) [2] y el *ICFES* vienen trabajando en lineamientos para la acreditación y estandarización básica de los currículos en Ingeniería. En este caso, se centrará la atención en el currículo de Ingeniería de Sistemas.

En 2003, *ACOFI* basado en estudios anteriores y en recomendaciones de *ACM*, presentó una propuesta [2] acerca de las áreas sobre las cuales deberían evaluarse los estudiantes de Ingeniería de Sistemas del país, lo cual además sirvió como una aproximación a los contenidos mínimos que deberían incluir todos los currículos de Ingeniería de Sistemas. En dicho documento se hace una división inicial de las áreas comunes a otras ingenierías y las áreas propias de la disciplina, y a su vez, ésta la subdivide en: Ciencias Básicas de Ingeniería y áreas de formación profesional; dentro de esta última se ubica la Ingeniería de Software y entre sus habilidades y destrezas, se encuentran el diagnosticar, diseñar, especificar, construir, evaluar, mantener y administrar proyectos de software.

Así mismo, *ACOFI* [2] plantea las competencias que deben desarrollar los Ingenieros de Sistemas: Deducir e interpretar datos e información relevantes, comunicarse clara y consistentemente, pensar analíticamente para la solución de problemas y tomar decisiones efectivas bajo presión.

Esto nuevamente conduce a la necesidad de establecer mecanismos pedagógicos que potencien el desarrollo de tales competencias en el área de Ingeniería de Software, concretamente en la etapa de Análisis y Diseño.

Como se observa, tanto a nivel nacional como internacional, es evidente que el proceso de Análisis y Diseño es fundamental en el desarrollo de Software y como tal, debe ser parte esencial de los contenidos asimilados por estudiantes de Ingeniería de Sistemas. Sin embargo, en la mayoría de las universidades

colombianas que ofrecen esta carrera, el Análisis y Diseño se dicta de manera conjunta con la parte de programación y/o Ingeniería de Software y dentro de esas asignaturas se hace evidente el poco tiempo que se dedica al Análisis y Diseño. Los mecanismos de enseñanza - aprendizaje son convencionales y no facilitan el desarrollo de las competencias requeridas en el área. Estas actividades convencionales no permiten a los estudiantes tener vivencias prácticas de proyectos de corto, mediano o largo alcance; adicionalmente, en Internet encuentran materiales o ejemplos aislados que no muestran de manera integral el proceso de análisis y diseño en toda su dimensión.

En la Pontificia Universidad Javeriana (sede Bogotá, Colombia), el currículo de Ingeniería de Sistemas tenía inmerso el *Análisis y Diseño Orientado a Objetos (ADOO)* dentro de la asignatura *Programación Orientada a Objetos (POO)* hasta el año 2005. Sin embargo, a partir del segundo semestre de dicho año se llevó a cabo la separación [14] de estas dos asignaturas. Inicialmente, los contenidos programáticos eran muy similares; no obstante, a medida que se ha venido impartiendo la nueva asignatura han surgido necesidades y temáticas que han generado grandes cambios a tal punto que hoy en día, las dos asignaturas aunque complementarias, tienen contenidos completamente diferentes. A lo largo de este proceso evolutivo, además de plantear nuevos contenidos, también se han desarrollado y ajustado herramientas didácticas encaminadas a facilitar en el estudiante, el desarrollo de destrezas y habilidades relacionadas con la solución de problemas prácticos.

Actualmente, *ADOO* hace énfasis en el modelado de los problemas a partir de los requerimientos del usuario y su posterior paso al modelo de artefactos de software como propuesta de solución a dicho problema.

### III. UNA NUEVA PROPUESTA PARA ADOO

Como se mencionó anteriormente, a partir del segundo semestre de 2005, se dio inicio a la asignatura de *Análisis y Diseño Orientado a Objetos (ADOO)*. Con este nuevo panorama, y dada la apertura a una nueva dinámica de flexibilidad en la Universidad, los estudiantes estaban en la libertad de cursar esta asignatura de manera simultánea, antes o después de la asignatura *Programación Orientada a Objetos (POO)*, lo cual implicaba que tales cursos fueran auto-contenidos a pesar de sus obvias relaciones.

En ese sentido se inició un proceso continuo de mejoramiento y ajuste de los contenidos de tales asignaturas, en especial, la relacionada con el tema de análisis y diseño, centro del tema del presente artículo. Este proceso ha sido impulsado en tres frentes: *i)* el seguimiento cuidadoso de los tiempos y el tema del curso, *ii)* el desarrollo de actividades de aprendizaje significativo y diseño de instrumentos de aprendizaje acorde con los objetivos del mismo y, *iii)* la consolidación del equipo de trabajo encargado del curso (ver Figura. 1).



Figura 1. Visión y frentes del proceso de mejoramiento de *ADOO*

#### A. Objetivos y contenido de la Asignatura

El objetivo general que persigue la asignatura es lograr que el estudiante esté en capacidad de analizar, diseñar e implementar soluciones a problemas usando el Paradigma Orientado a Objetos, sustentado bajo un esquema de proceso de Ingeniería de Software básico.

Para lograr este fin, *ADOO* cuenta con los siguientes objetivos específicos:

- Se pretende que el estudiante asimile los conceptos fundamentales de Análisis y Diseño Orientado a Objetos.
- Introducir al estudiante en los conceptos básicos del proceso de Ingeniería de Software especialmente de sus etapas iniciales de obtención de requerimientos, análisis y diseño bajo un enfoque Orientado a Objetos.
- Modelar gráficamente la solución de problemas con un enfoque Orientado a Objetos, usando un lenguaje de modelado, en este caso *UML*.

El contenido actual de la asignatura contempla tres fases (ver Figura. 2): la primera se centra en los conceptos básicos de los procesos de construcción de software; se utiliza una versión liviana y general del *RUP* (siglas de “*Rational Unified Process*”) como contexto de las actividades, documentos y personas involucradas en las etapas de análisis y diseño.

Posteriormente se hace énfasis en las actividades de análisis de casos reales, los principios de modelado utilizando *UML*, el levantamiento de requerimientos y la elaboración de casos de uso. Para estos propósitos se ha utilizado un instrumento de

---

1. RUP: <http://www-01.ibm.com/software/awdtools/rup/>

apoyo denominado *HACER & USOS*, el cual surgió como adaptación del instrumento GRACE [5]. Dicho instrumento es usado en las primeras etapas de una metodología general para el desarrollo de proyectos en cualquier ingeniería; los detalles al respecto se encuentran en la sección B correspondiente a las actividades e instrumentos de aprendizaje.

La segunda fase del curso se centra en el diseño de una solución computacional en la cual se detallan las actividades de modelado utilizando *UML*. Esta fase inicia con los conceptos fundamentales del paradigma Orientado a Objetos y continúa

con la construcción y propuesta de modelos de solución a los requerimientos identificados de la(s) situación(es) o caso(s) del mundo real.

La tercera parte del curso refina las actividades de modelado a través del uso de diversos talleres y laboratorios prácticos y la introducción de los conceptos de patrones *GRASP* (acrónimo de *General Responsibility Assignment Software Patterns*, Patrones de Principios Generales para Asignar Responsabilidades) [12] y algunos patrones *GoF* (acrónimo de *Gang of Four*, patrones de diseño) [8].

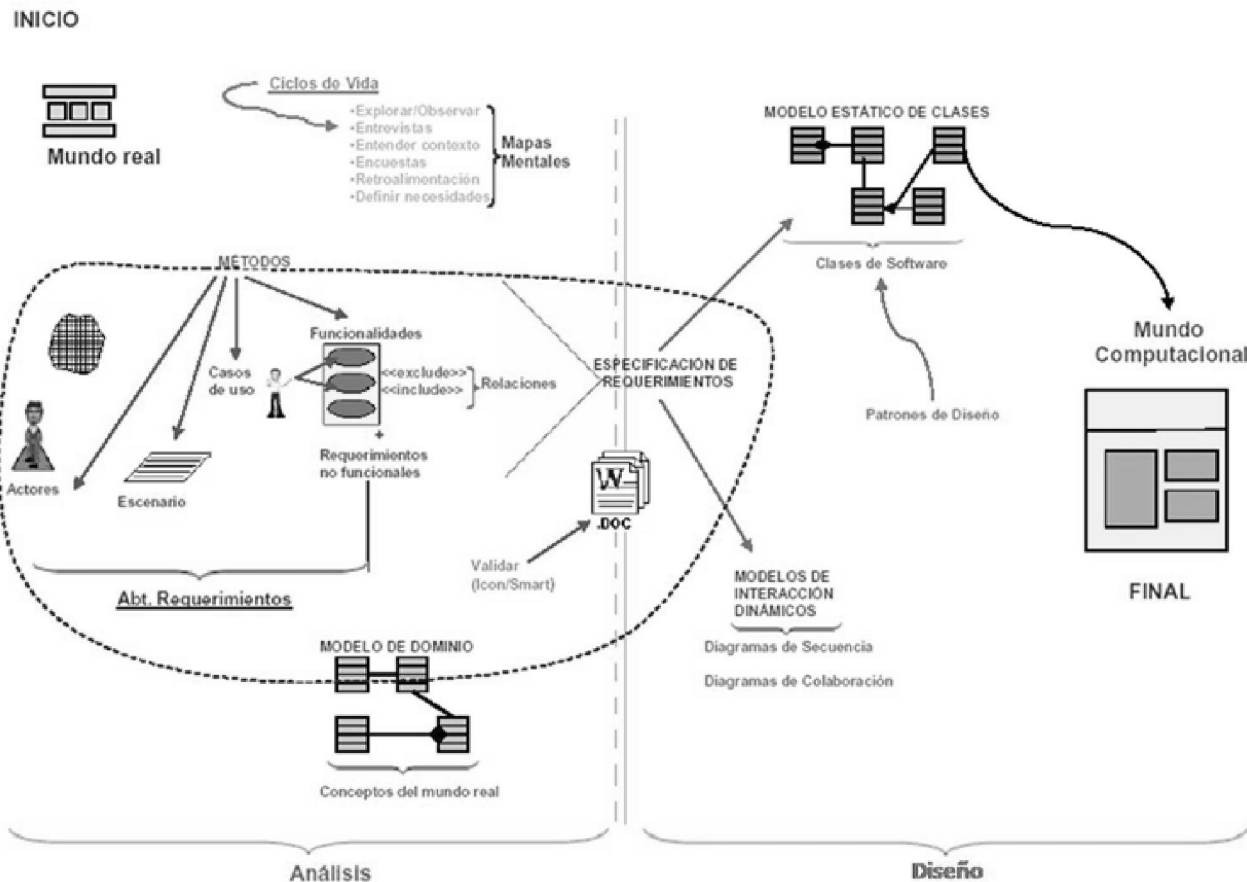


Figura 2. Percepción inicial del curso desde la perspectiva de uno de sus estudiantes<sup>2</sup>

## B. Actividades e Instrumentos de Aprendizaje

El grupo de trabajo involucrado en el desarrollo de la asignatura ha tenido presentes varios aspectos con respecto a algunos modelos pedagógicos que se fundamentan en el aprendizaje activo, en el modelo de pensamiento integral elaborado por Ned Herrmann [9] y en los estilos de aprendizaje

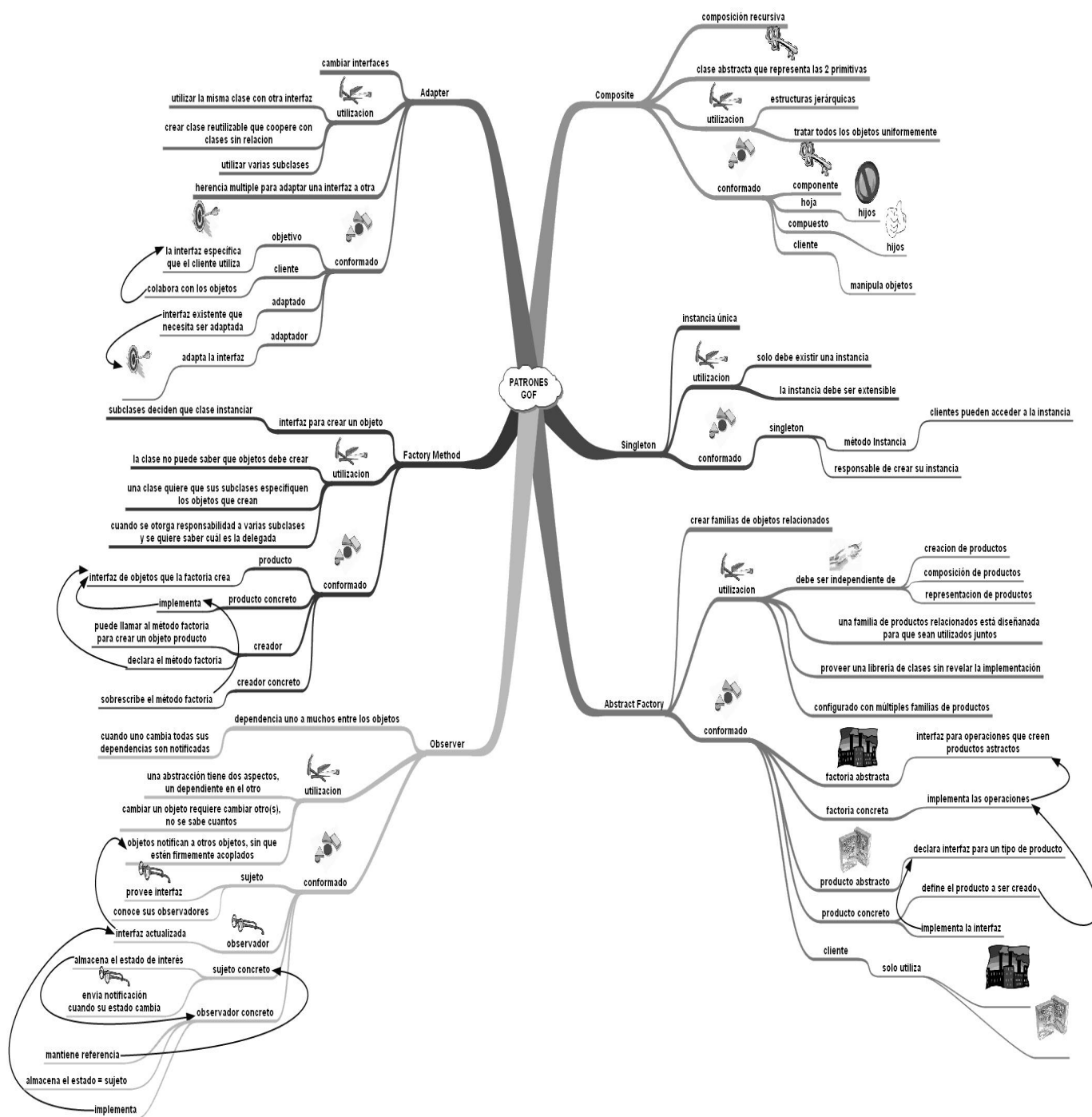
centrados en el estudiante ([1] y [13]) para el diseño de estrategias didácticas relacionadas con el desarrollo de actividades e instrumentos de aprendizaje; el detalle de estos aspectos se muestran en la sección IV.

A continuación se muestran de manera general las principales actividades de la asignatura:

**Sesiones de Clase:** La asignatura consta de 32 sesiones de clase en el semestre, tiempo en el cual se manejan varios ciclos de aprendizaje compuestos de lecturas previas de estudio de cada tema, elaboración de mapas mentales que demuestren la

2. Actividad de aprendizaje realizada a través de esquemas y resúmenes. Ejercicio presentado por la estudiante Camila Rojas durante el primer semestre de 2007.

comprensión y relación de dichos temas, desarrollo de talleres relacionados, prácticas de laboratorio y evaluaciones, entre otros.



**Figura 3.** Uso de mapas mentales (resumen de los patrones básicos de *GoF* impartido en la asignatura desde la perspectiva de uno de sus estudiantes)<sup>3</sup>.

3. Actividad de aprendizaje realizada a través de mapas mentales. Ejercicio presentado por la estudiante Maria Ronderos durante el segundo semestre de 2006

- **Proyecto:** De manera paralela a estos ciclos de aprendizaje, los estudiantes deben desarrollar un proyecto de curso que van refinando continuamente a medida que se avanza en cada uno de los temas. El enunciado del proyecto particular de cada semestre es puesto a punto para que tenga en cuenta, si no todos, la mayoría de los aspectos del curso. Se realizan dos exposiciones con entregas del material desarrollado para verificar los avances de los modelos generados, la documentación respectiva a través del instrumento *HACER & USOS* y la retroalimentación necesaria para asegurar la calidad del entregable final.
- **Actividades Prácticas:** Las actividades prácticas de taller o laboratorio se realizan semanalmente después de una sesión teórica. Estas actividades se centran en un conjunto de pequeños casos de estudio que se han establecido previamente por los docentes de la asignatura. Para cada práctica se hace entrega del enunciado de caso correspondiente, los estudiantes trabajan en una herramienta propietaria Together® o en software libre, particularmente *Dia*<sup>4</sup>. Una vez entregada la posible solución o modelo, se realizan observaciones de retroalimentación de manera directa por los docentes y el soporte de los monitores de la asignatura.

### C. Personas involucradas en el proceso

El recurso humano involucrado en el desarrollo de este proceso contempla, en primera instancia, a los mismos estudiantes, los monitores que brindan soporte a las actividades de aprendizaje y los docentes que guían toda la dinámica del curso.

- **Los estudiantes:** La mayoría de los estudiantes que ingresan a la asignatura están entre segundo y cuarto semestre de la carrera de Ingeniería de Sistemas. Ellos pueden cursar, durante el mismo semestre, tanto la asignatura de *ADOO* como la de *POO*, o pueden tomar *ADOO* en un semestre anterior o posterior al curso de *POO*. Cualquiera que sea el caso, el diseño auto-contenido de los cursos ha permitido que los estudiantes tengan la libertad de escoger el orden en el cual deseen tomar tales asignaturas. Dicha decisión por parte de los estudiantes, es apoyada por las sesiones de consejería que se realizan durante cada semestre en la universidad y de manera particular, para este caso, a través de los docentes del área de Ingeniería de Sistemas. En general, revisando la información académica de los estudiantes y tomando en cuenta sus opiniones sobre estas dos asignaturas, existe una tendencia (sesgada un poco por los procesos de consejería) en cursar primero *ADOO* y luego *POO*. Lo que sí es claro es asegurar que los temas de prerrequisito, asociados con pensamiento algorítmico y programación básica, estén bien fundamentados en los estudiantes. En este sentido, y dada la sinergia de este proceso,

se ha generado una mayor comunicación y alineación entre las asignaturas del área de construcción de Software que incluye, además de los estos cursos básicos de análisis, diseño y programación, los niveles de Ingeniería y Arquitecturas de Software dentro de la carrera de Ingeniería de Sistemas.

- **Los Monitores:** Un apoyo en el seguimiento y la retroalimentación de los avances de aprendizaje de los estudiantes se realiza a través de monitores de cada una de las secciones de la asignatura. Tales monitores son estudiantes de semestres superiores de la carrera que aportan su experiencia en el análisis, elaboración de modelos y diseños *OO* con la guía de los docentes.
- **Los Docentes:** En cuanto al grupo de docentes de la asignatura se ha establecido una dinámica de trabajo que involucra aspectos como los siguientes:
  - **Reuniones semanales** antes, durante y al final de cada semestre para ajustar los detalles que sean del caso, acorde con el avance de los temas en cada una de las secciones de la asignatura.
  - **Elaboración de las actividades y los instrumentos de aprendizaje** en consenso. Gran parte de este material es publicado de manera abierta en las páginas web de los docentes o a través de la plataforma (Blackboard ®) utilizada en la Universidad para la administración de tales recursos.
  - **Elaboración y definición de criterios de evaluación** de las actividades e instrumentos utilizados en clase, en especial de las pruebas establecidas que permiten valorar el avance de los estudiantes. A manera de ejemplo, los parciales del curso se elaboran en consenso, se establece la solución esperada, el tiempo previsto para la solución y la concordancia con los temas planeados en cada corte del semestre.
  - **Calificación y corrección de las pruebas establecidas** (laboratorios, talleres, quices y parciales) y en general del material didáctico utilizado en clase con el apoyo de los monitores asignados y la utilización de Blackboard®
  - **Ajustes y evaluación de los contenidos** del curso así como el seguimiento de las actividades e instrumentos relacionados con las sesiones de clase. Como detalle especial de este punto, se han llevado a cabo **bitácoras de avance** de las secciones para ser utilizadas como insumo con el fin de analizar los ajustes en el avance de los temas, las mejoras para el siguiente semestre y como guía pertinente para cada profesor.

Esta dinámica ha permitido asegurar una experiencia proactiva y enriquecedora en el tema así como una visión a futuro de un mayor fortalecimiento del tema, el refinamiento de las actividades e instrumentos didácticos que motiven y estimulen el aprendizaje, así como la construcción de un banco de proyectos que se podrá abrir a la comunidad académica interesada en el área, como uno de los proyectos a corto plazo del grupo de trabajo.

4. DIA: <http://projects.gnome.org/dia/>

#### IV. EXPERIENCIA SOBRE EL APRENDIZAJE ACTIVO ¿CASO EXITOSO?

En esta sección se presenta la experiencia de un semestre típico de *ADOO*. Las diversas actividades (parciales, talleres, proyecto, etc.) conforman lo que se ha denominado aprendizaje activo donde la mayor parte de la labor educativa recae en el estudiante.

Semana a semana el curso se divide en dos partes: primero, una sesión teórica para la cual los estudiantes preparan los temas a través de lecturas predefinidas y a partir de éstas, ellos elaboran mapas mentales con el fin de comprender los diversos conceptos consignados en cada uno de los temas. Por otro lado, se tiene una sesión práctica en la cual los estudiantes frente a computadores, practican los temas vistos en la sesión teórica. En esta sesión práctica se utiliza la herramienta *HACER & USOS* (variación de *GRACE* [5] para la parte de levantamiento y documentación de requerimientos de un proyecto de Ingeniería de Sistemas) y una modificación de las plantillas de *Cockburn* [7] (incorporada en *HACER & USOS* con el fin de documentar los requerimientos). Adicionalmente generan los modelos relacionados con la etapa de análisis y diseño utilizando *UML* (por ejemplo, diagramas de casos de uso, de clases, de secuencia).

En la sesión teórica, el docente aclara dudas a los estudiantes y les entrega diversos ejercicios que les brindan una idea de cómo aplicar los diversos conceptos. La mayoría de estos ejercicios se desarrollan en grupos de dos o tres personas y entre todos se propone una posible solución. Otro enunciado es entregado a los estudiantes con el fin de revisar y preparar la sesión práctica que se desarrollará durante la siguiente clase.

Para el proyecto, los docentes preparan un enunciado antes de que comience el semestre académico. Allí se concentran la aplicación de los conceptos fundamentales de la asignatura. Este proyecto se desarrolla en dos etapas: primero se levantan y documentan los requerimientos de dicho proyecto a través de la adaptación de la plantilla de *Cockburn* [7]. Además, se utiliza la herramienta *HACER & USOS* y el correspondiente diagrama de

Casos de Uso. Adicionalmente se solicita el modelo de dominio de tal enunciado. La segunda parte corresponde al modelo estático: el diagrama de clases en las que no sólo se asignan los atributos y métodos sino las responsabilidades de cada una de las clases a través de la aplicación de los patrones *GRASP* y *GoF*. Finalmente se modela la parte dinámica del proyecto a través de los diagramas de colaboración y de secuencia.

Este proceso incremental de las entregas del proyecto se asemeja al desarrollo de la parte práctica que se hace durante los tres exámenes parciales aplicados a los estudiantes. Además, todas las entregas tienen una pre-sustentación en la que se detectan algunos errores en los modelos o entregables y son los docentes y los compañeros de los estudiantes que presentan, quienes brindan retroalimentación al respecto. De igual manera, se pretende detectar problemas comunes y que haya un aprendizaje colectivo tanto a la hora de hacer una corrección como para detectar ellos mismos los errores que pueden surgir

#### V. ACTIVIDADES E INSTRUMENTOS PARA EL APOYO DE LOS PROCESOS DE ENSEÑANZA/APRENDIZAJE

Con el fin de apoyar los procesos de Enseñanza/Aprendizaje en *ADOO* se han utilizado y diseñado diversas actividades e instrumentos entre los cuales se pueden mencionar:

- **Creación y Diseño de Casos de Estudio:** A lo largo de los últimos dos años se han adaptado y redactado casos de estudio para que los estudiantes, a medida que transcurre el curso, los vayan desarrollando. Cada semana se hacen talleres donde progresivamente los estudiantes aplican los temas vistos en clase. Además, los casos de estudio se han diseñado con diferentes niveles de complejidad para ser evaluados en parciales así como en el proyecto del curso. Para cada caso de estudio se ha creado un plan de trabajo donde se especifican los entregables con sus respectivas fechas (ver Figura. 4) y además, aparece el cuerpo del enunciado (ver Figura. 5):

Entregas	Entregables / Productos	Fecha Máxima de Entrega
Primera	<ul style="list-style-type: none"> <li>Diligenciamiento del formato HACER-S (libro Excel)</li> <li>Diagrama de Casos de Uso (imagen jpg)</li> <li>Especificación (plantilla adaptada de Cockburn) de 10 casos de uso</li> <li>Diagrama de clases aplicando /identificando patrones GRASP (imagen jpg)</li> </ul>	<ul style="list-style-type: none"> <li>Versión Digital e Impresa: Sesión 20 de clase <ul style="list-style-type: none"> <li>1 de octubre (sección LCD)</li> <li>2 de octubre (sección DA)</li> <li>3 de octubre (sección AC)</li> </ul> </li> </ul>
Segunda	<ul style="list-style-type: none"> <li>Diagrama de clases aplicando /identificando patrones GoF (imagen jpg)</li> <li>Análisis del uso de patrones</li> <li>5 diagramas de secuencia</li> <li>5 diagramas de colaboración</li> </ul>	<ul style="list-style-type: none"> <li>Versión Digital e Impresa: Jueves 20 de noviembre hasta las 6pm.</li> </ul>

Figura 4. Plan de trabajo de un proyecto planteado para un caso de estudio.

## ENUNCIADO



Una prestigiosa empresa de turismo llamada *ADOO-Turismo* ha contratado sus servicios para modelar un sistema de manejo de paquetes y planes turísticos para destinos tanto nacionales como internacionales.

Esta empresa ofrece diversos tipos de productos: tiquetes aéreos, hoteles, renta de vehículos, planes y paquetes para una persona o para un grupo. De igual manera ofrece paquetes especiales para familias. Un paquete de grupo podría considerarse conformado por varios paquetes de una sola persona, varios paquetes de familia o a su vez, varios paquetes de grupo.

A la hora de escoger un destino, el sistema le pregunta al usuario qué necesita: tiquete, hotel, transporte interno, carro rentado, etc. Por ejemplo, si el cliente sólo quiere tiquetes aéreos, el sistema se conecta con los sistemas de las aerolíneas y le ofrece la mejor opción a los usuarios dependiendo de sus necesidades: el tiquete más económico,

**Figura 5.** Fragmento de un enunciado del proyecto planteado a partir de un caso de estudio.

- **Temario de parcial:** donde la parte práctica muestra una progresión en el desarrollo del tema. En la asignatura se han planteado tres parciales con dos componentes importantes: uno conceptual y otro práctico. Para este último se crea un enunciado, como se explicó en el ítem anterior, y se va enriqueciendo, teniendo en cuenta los temas que deben ser evaluados (es decir, para el caso de estudio, es redacta entorno al mismo tema para los tres parciales pero los aspectos a evaluar son diferentes).
- **Adaptación de la plantilla HACER-ICON** [5] llamada *HACER & USOS* (ver Figura. 6), es utilizada para cubrir las necesidades del tema de Análisis OO. Además, se adicionó a *HACER & USOS* una simplificación de la plantilla para documentación de casos de uso propuesta por Cockburn [7] (ver Figura. 7):



**Figura 6.** Plantilla en Excel *HACER & USOS* que permite documentar la parte de análisis de un proyecto.



Proyecto:		Fecha:	
Autor:		Versión:	

<b>Id Caso de Uso:</b>		<b>Nombre:</b>	
<b>Objetivo en Contexto (Resumen):</b>			
<b>Actores Participantes</b>			
<b>Entradas</b>			
<b>Salidas</b>			
<b>Pre-Condiciones</b>			
<b>Post-Condiciones</b>		Condición final de éxito:	
		Condición final de fallo:	

Flujo básico de éxito			
No.	Actor	No.	Sistema

<b>Variaciones (Caminos Alternativos):</b>	
<b>Variaciones (Caminos de excepción):</b>	
<b>Extensiones</b>	
<b>Requerimientos Asociados</b>	

Figura 7. Simplificación de la plantilla de Cockburn.

- Actualmente se está elaborando un **Banco de proyectos** en el cual se consignan los enunciados de los casos de estudio con una posible solución (compuesta de todos los diagramas y documentos de Análisis y Diseño) para que los estudiantes tengan un conjunto de casos resueltos por los que se puedan guiar para desarrollar sus trabajos, tareas, proyectos, etc. Así mismo, los profesores se pueden guiar para explicar los diversos temas, teniendo en cuenta las soluciones propuestas.
- **Diapositivas** de todas las clases como apoyo a los docentes y como guía para los estudiantes (ver Figura. 8).

Home | Docencia | Investigación | Consejería | Otros intereses

---

**Docencia**

Generalidades

ADOO 2008-3

ADOO 2008-1

ADOO 2007-3

ADOO 2007-1

ADOO 2006-3

ADOO 2006-1

Ing Software 2007-3

Ing Software 2007-1

Ing Software 2006-3

Ing Software 2006-1

HCI 2008-1

**Investigación**

Generalidades

Software Tangible

Trabajos de Grado

4 / 13

51,9%

Buscar

**¿Qué es un actor?**

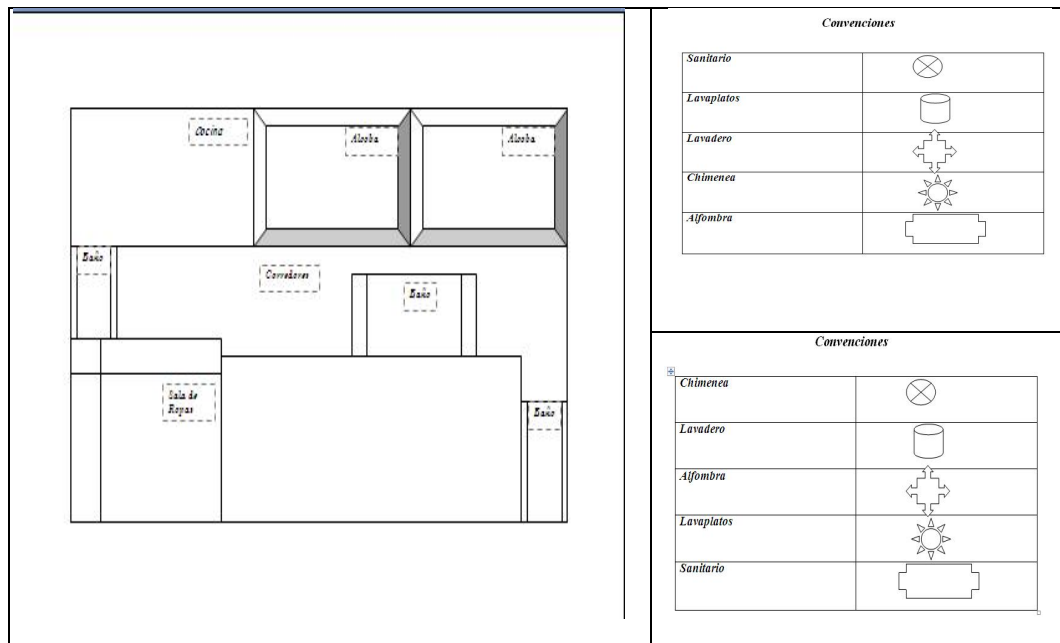
- Representa un conjunto coherente de roles que juegan los usuarios de los casos de uso
- Pueden ser sistemas automáticos o personas.

nombre Actor

Figura 8. Diapositivas de los temas de ADOO.

- **Contextualización de diversos temas** (por ejemplo, importancia del modelado y levantamiento de requerimientos) a través de talleres prácticos cuyo objetivo es mostrar necesidades tales como el diseño de un apartamento utilizando convenciones unificadas (para que todos entiendan lo mismo, ver Figura. 9.) o los requerimientos cambiantes de alguien que solicita la

construcción de un avión de papel. Todo esto se hace para mostrarle a los estudiantes que es necesario un proceso anterior a lanzarse a construirlo. Muestran además, la necesidad de comunicarse con el cliente/usuario y el importante rol de éste para que el producto se acomode a sus requerimientos.



**Figura 9.** Modelado de un apartamento utilizando diversas convenciones con el fin de mostrar la importancia del modelado unificado. Note que cada convención significa diferente en cada conjunto.

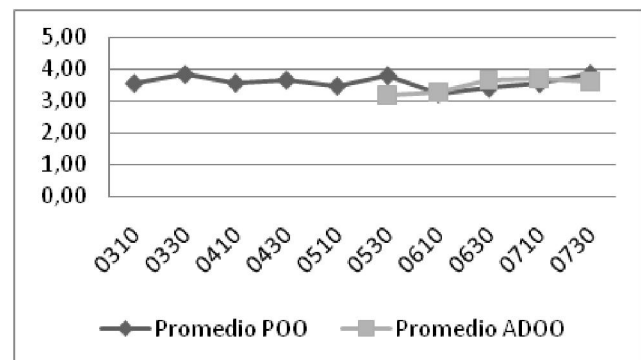
- De igual manera se ha hecho una selección de temas provenientes de diversos libros que pueden encontrarse en la bibliografía del curso, con el fin de concentrarse en los conceptos de base que sus autores proponen y luego desarrollar los casos de estudio elaborados especialmente para *ADOO*. Dichas lecturas son evaluadas a través de comprobaciones de lectura y mapas mentales [6] (ver descripción de Mapas Mentales en la sección III.B). Tales mapas mentales sirven para sintetizar los temas y con ellos se ha demostrado que los estudiantes entienden más y mejor los conceptos consignados en las lecturas; de igual manera, se han mejorado los resultados obtenidos tanto en las comprobaciones de lectura, como en la parte conceptual del parcial.

## VI. DESCRIPCIÓN DE LAS EVALUACIONES DEL CURSO

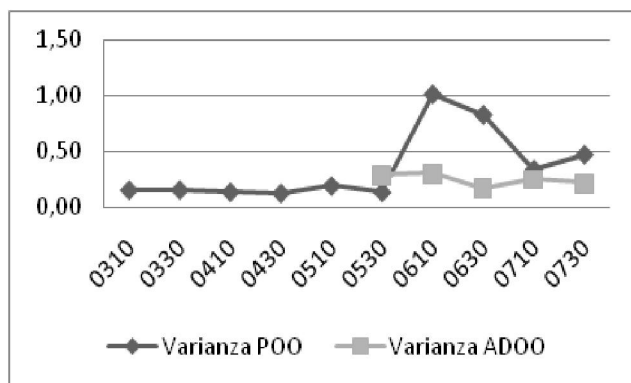
En esta sección se muestran brevemente algunos datos descriptivos del comportamiento del desarrollo de la asignatura a través de las notas obtenidas por parte de los estudiantes en un lapso de cinco semestres.

### A. Relación entre ADOO y POO

En términos de tiempo, el curso de *POO* se ha dictado durante once semestres mientras el de *ADOO* se ha dictado durante seis semestres. Las Figuras 10 y 11 muestran el promedio obtenido por los estudiantes desde el primer semestre de 2003 hasta el segundo semestre de 2007.



**Figura 10.** Promedios de los cursos de *ADOO* y *POO*.



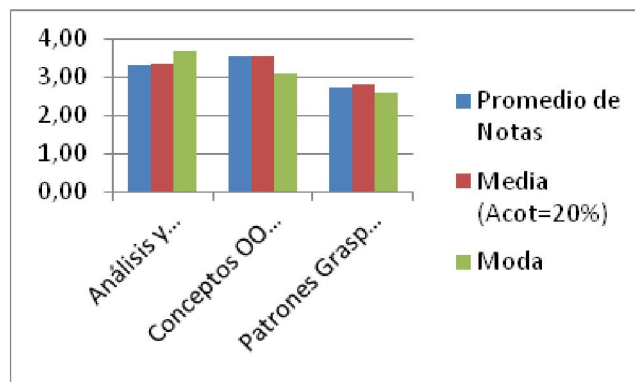
**Figura 11.** Varianza en los promedios de los cursos de *ADOO* y *POO*.

Se puede indicar que el inicio del curso de *ADOO* marcó lógicamente cambios y reformas en el curso de *POO*, lo que generó una desestabilización en su desarrollo y en su evaluación asociada con la manera tradicional en la que se venía realizando. Ese fenómeno se ha estabilizado dentro de la evolución de los cursos, sus contenidos y el diálogo entre sus docentes. Actualmente uno de los docentes realiza sus actividades en las dos asignaturas (*ADOO* y *POO*) lo que permitirá seguir un proceso de mejoramiento continuo.

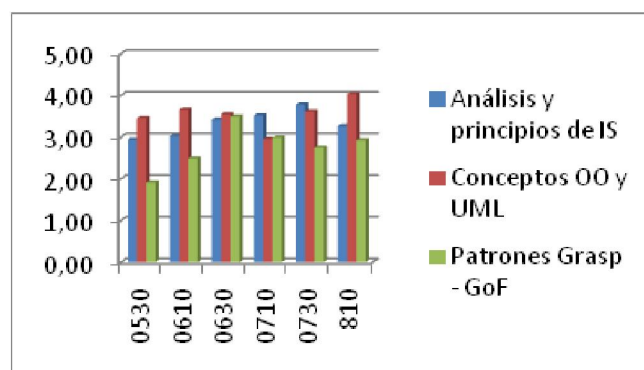
### B. Evolución de los temas del curso

A pesar de la realización de ciertos ajustes en los contenidos del curso, se han mantenido tres bloques de temas a saber: el Análisis Orientado a Objetos enmarcado dentro los fundamentos de Ingeniería de Software, los conceptos básicos del Paradigma Orientado a Objetos y *UML* y, finalmente, una introducción a los patrones *GRASP* y *GoF*. En las Figuras 12 y 13 se puede verificar que la aplicación del aprendizaje activo, como se ha descrito en este artículo, se ha centrado principalmente en las dos primeras partes del curso a través de las actividades, instrumentos y técnicas ya descritas. El desarrollo futuro del banco de proyectos mencionado anteriormente permitirá consolidar el tema de patrones.

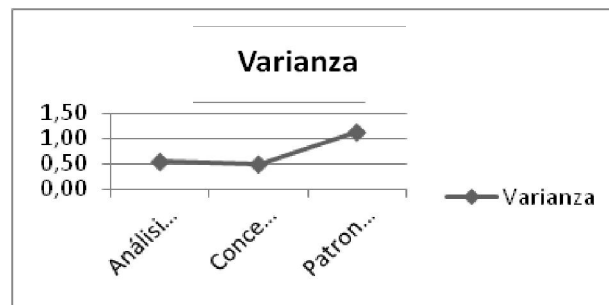
Los datos permiten observar que el tema de los conceptos centrales de orientación a objetos junto con las actividades de análisis y modelado utilizando *UML* se destacan y se mantienen dentro de la evolución de la asignatura. Vale la pena anotar que la varianza de las notas es mayor en el tema de patrones (ver Figura 14).



**Figura 12.** Evaluación promedio de los temas de *ADOO* (desde el segundo semestre de 2005 y hasta el primer semestre de 2008)



**Figura 13.** Evaluación promedio por semestres de los temas de *ADOO*



**Figura 14.** Varianza de los promedios de las notas en la evaluación de los temas de *ADOO*

## VII. CONCLUSIONES Y TRABAJO FUTURO

En este artículo se ha presentado la evolución que ha tenido la asignatura Análisis y Diseño Orientada a Objetos (*ADOO*) en una institución universitaria. En dicha evolución se ha separado completamente la parte de Análisis y Diseño de la parte de Programación. Para este propósito se han utilizado diversas estrategias, herramientas, instrumentos de aprendizaje y

actividades basadas en el Aprendizaje Activo con lo cual, el papel del estudiante es uno de los principales en este proceso de enseñanza/aprendizaje. Adicionalmente, este material de apoyo ha servido como guía para los docentes de la asignatura, creando nuevas perspectivas de constante mejoramiento no sólo en asignaturas relacionadas con el Paradigma Orientado a Objetos, sino en aquellas que tienen como prerrequisito las asignaturas en cuestión.

Finalmente se presentaron los resultados de esta nueva propuesta de *ADOO* a través de una revisión general de estadística descriptiva que permite visualizar el éxito de los estudiantes en la evaluación de los conceptos adquiridos de orientación a objetos, análisis y modelado *UML* sin realizar actividades de programación directamente asociada.

Como perspectivas de este trabajo se tiene el mantenimiento del banco de proyectos y el ajuste de los instrumentos, actividades y estrategias de apoyo al proceso de enseñanza/aprendizaje. Así mismo, se plantea el diseño de nuevas asignaturas en las que se pueda aplicar la experiencia adquirida en *ADOO*.

#### REFERENCIAS

- [1] Allinson, C. y Hayes, J., 1996. The Cognitive Style Index: A measure of intuition-analysis for organizational research. *Journal of Management Studies*.
  - [2] Asociación Colombiana de Facultades de Ingeniería, 2005. Marco de Fundamentación Conceptual: Especificaciones De Prueba: Ecaes Ingeniería De Sistemas Versión 6.0.
  - [3] Association for Computing Machinery (ACM), 2004. Association for information Systems (AIS), Computer Society (IEEE-CS): Computing Curricula 2004: The Overview Report covering undergraduate degree programs in: Computer Engineering, Computer Science, Information Systems, Information Technology, Software Engineering. ACM & IEEE Computer Society. U.S.A.
  - [4] Association for Computing Machinery (ACM), 2005. Association for information Systems (AIS), Computer Society (IEEE-CS): Computing Curricula 2005: The Overview Report covering undergraduate degree programs in: Computer Engineering, Computer Science, Information Systems, Information Technology, Software Engineering. ACM & IEEE Computer Society. U.S.A.
  - [5] Barros, R.; Duque, G.; Rojas, J.; Sánchez, L. y Velosa, J., 2005. Introducción a la Ingeniería. EAN.
  - [6] Buzan T., 1996. El libro de los mapas mentales – Cómo utilizar al máximo las capacidades de la mente. Barcelona: Ediciones Urano SA.
  - [7] Cockburn, A., 1997. Using Goal-Based Use Cases. En: *Journal of Object-Oriented Software Engineering*, Vol. 10, No. 7, pp. 56-62.
  - [8] Gamma, E., Helm, R., Johnson, R., y Vlissides, J. 1995. Design Patterns, Addison-Wesley.
  - [9] Herrmann, N., 1996. The whole brain business book. McGraw Hill, New York.
  - [10] Instituto Colombiano para el Fomento de la Educación Superior. <http://www.icfes.gov.co/>
  - [11] Kaiser, A., 2001. Directrices para el desarrollo de los nuevos currículos de TIC: Pensamiento Sistémico y formación. En: Directrices para el desarrollo curricular: Nuevos currículos de TIC para el siglo XXI: el diseño de la educación del mañana. Career Space, Oficina de Publicaciones Oficiales de las Comunidades Europeas, Luxemburgo. pp. 27.
  - [12] Larman, C., 2002. UML y Patrones. Segunda Edición, Pearson Educación.
  - [13] Palloff, R. y Pratt, K., 2003. The Virtual Student. Jossey-Bass.
  - [14] Pontificia Universidad Javeriana Departamento De Ingeniería De Sistemas, Sección de Programación y Desarrollo de Software, <http://sophia.javeriana.edu.co/programacion/>
  - [15] Revill, M., 2001. Nota sobre la formación situacional la adquisición explícita de capacidades conductuales. En: Directrices para el desarrollo curricular: Nuevos currículos de TIC para el siglo XXI: el diseño de la educación del mañana. Career Space, Oficina de Publicaciones Oficiales de las Comunidades Europeas, Luxemburgo. pp. 28.
- Deicy Alvarado.** Ingeniera de Sistemas de la Universidad Distrital Francisco José de Caldas, Bogotá, Colombia (1991). Especialista en Multimedia Educativa de la Universidad Antonio Nariño, Bogotá Colombia (1995). Magister en Ingeniería de Sistemas de la Universidad Nacional de Colombia, Bogotá Colombia (2003). Doctor en Ciencias de la Computación e Inteligencia Artificial Universidad de Oviedo, Gijón-Asturias España (2002). Profesora de Cátedra de la Universidad Distrital (1993-1995) y desde Abril de 1995 es Profesora de Planta de la misma Universidad. En 1993 planteó la creación del Grupo de Investigación en Informática Educativa al interior de la Universidad Distrital (*GIE-UD*) el cual dirige actualmente. Su trabajo se ha enfocado en la aplicación de la Informática en la Educación, la utilización de técnicas de Inteligencia Artificial en la solución de diversos problemas y el Análisis Algorítmico.
- Angela Carrillo Ramos.** Ingeniera de Sistemas y Computación (1996) y Magister en Ingeniería de Sistemas y Computación (1998) de la Universidad de los Andes, Bogotá, Colombia. Doctorado en Informática de la Universidad Joseph Fourier, Grenoble, Francia (2007). Asistente de Investigación y Profesora de Cátedra de la Universidad de los Andes (1996-1997). Profesora Asistente de la Universidad de los Andes (1998-2003). Desde Julio de 2007 es Profesora Asociada e Investigadora de los grupos *ISTAR* (el cual dirige) y *SIDRe* de la Pontificia Universidad Javeriana de Bogotá. Su trabajo se ha enfocado en el acceso a Sistemas de Información a través de dispositivos móviles utilizando la tecnología de agentes. Otros de sus intereses son la Adaptación (Personalización) de la información en ambientes nómadas, la construcción de Software y la Educación.
- Luis Carlos Díaz Chaparro.** Ingeniero de Sistemas de la Universidad Nacional de Colombia. Magister en Ingeniería de Sistemas y Computación de la Universidad de los Andes y Especialista en Creación Multimedia de la Facultad de Artes de la misma universidad. Se ha desempeñado en varios cargos de dirección universitaria; ha hecho parte de diversos grupos de trabajo en investigación y desarrollo tecnológico. Además, ha realizado estudios en el área de formación pedagógica, diseño curricular e investigación. Docente durante 12 años en diferentes universidades en áreas de programación, ingeniería de software, desarrollo multimedia, calidad de software, programación en la Web, Interacción hombre-máquina, entre otras. Actualmente se desempeña como Profesor Investigador en el Departamento de Ingeniería de Sistemas de la Pontificia Universidad Javeriana y es el Director de la Carrera de Ingeniería de Sistemas. También fue Director de la Especialización en Arquitectura Empresarial de Software de la Facultad de Ingeniería, director y jurado de trabajos de grado en el área de educación, multimedia, ingeniería de software y arquitectura de software. Miembro del grupo de investigación *ISTAR* y líder del programa de investigación "*Software Tangible*" relacionado con el estudio, identificación y propuesta de representaciones visuales, metáforas y modelos para la construcción de material didáctico que contribuyan al proceso de enseñanza aprendizaje de programación de computadores.