

Un modelo de diseño curricular de Informática orientado a la obtención de competencias

Félix Buendía, Juan-Carlos Cano, Julio Sahuquillo, Juan-Luis Posadas, Juan-Miguel Martínez, José-Vicente Benlloch

Departamento de Informática de Sistemas y Computadores
Escuela Técnica Superior de Informática Aplicada
Universidad Politécnica de Valencia

e-mail: {fbuendia, jucano, jsahuqui, jposadas, jmmr, jbenlloch}@disca.upv.es

Resumen

La mayoría de propuestas de diseño curricular en las titulaciones universitarias de Informática se han centrado en la definición de contenidos sin prestar demasiada atención a los aspectos metodológicos y su aplicación a la enseñanza de los elementos curriculares. Asimismo, se observa una falta de definición de las competencias obtenidas tras cursar dichas titulaciones. Con el fin de paliar estas carencias, este trabajo presenta una propuesta de diseño curricular en el área de Informática que organiza la enseñanza de dicha disciplina mediante un modelo orientado a la obtención de competencias.

1. Introducción

La universidad española está inmersa en un proceso de cambios derivados de la implantación de la LOU [1] y de acontecimientos como la Declaración de Bolonia [2]. Los estudios universitarios de Informática no son ajenos a estas circunstancias y actualmente se encuentran bajo una serie de revisiones para actualizar tanto sus contenidos y “curricula” como los métodos y procedimientos para su impartición.

Por un lado, los planes de estudio de Informática en niveles universitarios se han centrado en la definición de “curricula” para las diversas titulaciones sin prestar demasiada atención a los aspectos metodológicos que incorpora un modelo de diseño curricular. En este sentido, el diseño curricular según Sarramona [3] consiste en “la manera como se encarga la planificación de los elementos y procesos del currículo; esto es, la forma de organizar las enseñanzas de una disciplina

académica, de acuerdo con un determinado modelo didáctico”.

Por otro lado, se observa en la redacción de los planes de estudio en titulaciones de Informática, una falta de definición de las competencias obtenidas tras cursar dichas titulaciones.

Este trabajo presenta una propuesta de diseño curricular en el área de Informática que contempla tanto los aspectos curriculares como los metodológicos. El objetivo principal de la propuesta consiste en relacionar los elementos y procesos del currículo con la definición y obtención de competencias en el ámbito informático.

La mayoría de propuestas de diseño curricular (por ejemplo, en forma de proyectos docentes como el propuesto por Escudero [4]) en las titulaciones universitarias de Informática evidencian una separación entre los contenidos que forman parte del currículo y el enfoque o modelo docente para desarrollar su enseñanza. Dicha separación se deriva de la propia definición de los planes de estudio que rigen tales titulaciones. Por ejemplo, la resolución que establece el plan de estudio del Ingeniero Técnico en Informática de Sistemas de la UPV [5] indica una relación de materias troncales que contiene asignaturas, la distribución de créditos por cada asignatura, una breve descripción de contenidos (descriptores) y la vinculación a las áreas de conocimiento. Sin embargo, se observan pocas referencias al método o forma de organizar o gestionar la enseñanza de dichas materias.

Esta situación se repite en el planteamiento de los planes de estudio de informática en otros países como México [6] o Brasil [7]. Algunas universidades como la Autónoma de Madrid [8] hacen referencia a “una importante componente

aplicada en las enseñanzas impartidas”. Además indican que el plan de estudios “cualifica a sus titulados para cubrir las necesidades de la sociedad de la información en todas las áreas relacionadas con la informática” e incluso afirman que “Todos nuestros titulados han encontrado trabajo dentro del sector informático y la mayoría han podido elegir entre varias ofertas”.

Frente a esta situación, es posible encontrar modelos de diseño curricular que integran mecanismos para guiar la enseñanza de materias informáticas e incluso, que adecuan dicha enseñanza a las competencias que un alumno debe adquirir. El modelo más próximo es el utilizado en los Ciclos Formativos de Grado Superior (CFGs) en el área de Informática. Por ejemplo, el título de Administración de Sistemas [9] define planes de formación para adquirir determinadas competencias o capacidades profesionales. Para ello establece no sólo los contenidos que forman parte de la titulación sino también los objetivos, actividades y criterios de evaluación que intervienen en su enseñanza.

La propuesta de CC-2001 de ACM/IEEE [10] hace también referencia a los conocimientos y habilidades que debe adquirir un titulado en Informática. Otros ejemplos de definición de competencias se incluyen en titulaciones próximas a Informática como Biblioteconomía [11] que plantea un listado exhaustivo de competencias específicas como parte de una propuesta de diseño de titulación de grado en Biblioteconomía y Documentación. Dichas referencias muestran la importancia de definir competencias en ámbitos de docencia de la Informática y su utilidad para guiar la enseñanza en dichos ámbitos.

El resto del trabajo está estructurado en los siguientes apartados. La sección 2 describe el modelo de diseño curricular utilizado para organizar la enseñanza de Informática en titulaciones universitarias. La sección 3 muestra un ejemplo práctico de aplicación del modelo a una disciplina concreta relativa a los sistemas operativos. Por último, en la sección 4 se presentan las conclusiones y trabajos futuros.

2. Modelo de diseño curricular

El modelo de diseño curricular que se propone en este trabajo, está basado en la estructura de

Documento Base de Título (DBT) utilizado para definir titulaciones de Ciclos Formativos [12]. El DBT determina las enseñanzas mínimas para alcanzar una cierta competencia así como las condiciones en que se producen dichas enseñanzas.

DBT	
Unidades de competencia	Módulos
Realizaciones	Capacidades terminales
Criterios de realización	Criterios de evaluación
Dominio profesional	Contenidos mínimo

Figura 1. Estructura del Documento Base de Título

La Figura 1 muestra la estructura del DBT compuesta por Unidades de Competencia y Módulos. Dentro de una Unidad de Competencia se definen los siguientes elementos:

- Realizaciones: indican las tareas concretas en el ámbito de una cierta competencia.
- Criterios de Realización: permiten valorar el grado de cumplimiento de la Realización.
- Dominio Profesional: establecen el ámbito del trabajo relacionado con una cierta competencia.

Los Módulos se asocian a las Unidades de Competencia. Estos se organizan en los siguientes elementos:

- Capacidades Terminales: expresan los resultados que deben ser alcanzados por los alumnos.
- Criterios de Evaluación: permiten medir los resultados alcanzados en el Módulo.
- Contenidos: establecen los elementos que integran el Módulo.

2.1. Ejemplo de diseño curricular

A continuación se describe un ejemplo con las componentes del diseño curricular para un DBT en Informática. Una Unidad de Competencia puede consistir en “implantar y administrar sistemas informáticos en entornos multiusuario”. Una Realización asociada a dicho ejemplo sería “organizar y administrar la utilización de los recursos del sistema” mientras que un posible Criterio de Realización consistiría en “la forma de controlar los procesos en ejecución”. Un ejemplo de Módulo relacionado con la anterior Unidad de Competencia sería el de “Sistemas Informáticos Multiusuario”. Un caso de Capacidad Terminal

obtenida en el Módulo consistiría en “utilizar los recursos del sistema realizando funciones de usuario sobre el sistema operativo” y un posible Criterio de Evaluación sería la disponibilidad para “utilizar funciones del interfaz del sistema”.

2.2. Unidades de Trabajo

A partir de los elementos que configuran un Módulo se define un Diseño Curricular basado en Unidades de Trabajo que constan de un conjunto de objetivos, contenidos, actividades y una planificación temporal. En la Figura 2 se muestra un ejemplo de tabla para representar parte de los componentes de una Unidad de Trabajo. Dicha Unidad forma parte del Módulo “Sistemas Informáticos Multiusuario” y tiene una duración estimada de 20 horas.

Módulo profesional: Sistemas Informáticos Multiusuario			
Unidad de trabajo: Gestión de procesos Unix Tiempo: 20 h			
Conceptos/ Actitudes	Procedimientos	Actividades de aprendizaje	Actividades de evaluación
<ul style="list-style-type: none"> - Procesos en Unix - Ordenes en Unix - Interprete de órdenes - Llamadas a sistema de gestión de procesos 	<ul style="list-style-type: none"> - Lanzamiento de órdenes - Ejecución de procesos - Diseño de intérprete de órdenes 	<ul style="list-style-type: none"> - Ejercicios de llamadas a sistema - Monitorización de procesos - Resolución de problemas en el diseño de un intérprete de órdenes 	<ul style="list-style-type: none"> - Realizar e interpretar programas con llamadas a sistema - Revisar el resultado de la ejecución de procesos

Figura 2. Componentes de una Unidad de Trabajo

Los contenidos conceptuales de la Unidad están relacionados con la gestión de procesos y órdenes en UNIX y los contenidos procedurales hacen referencia al manejo de dichos procesos y órdenes. También se especifican dos tipos de actividades: las de aprendizaje, encaminadas a la resolución de ejercicios, prácticas y problemas relativos a los contenidos conceptuales y procedurales, y las de evaluación, que comprueban el grado de aprovechamiento.

La Unidad de Trabajo especificada constituye un ejemplo de elemento básico que serviría para estructurar y organizar un Diseño Curricular en una titulación de Informática. El mecanismo que completa el modelo propuesto consiste en asignar una Capacidad o conjunto de Capacidades Terminales a cada Unidad de Trabajo. Este mecanismo permite especificar su comportamiento o interacción con otras Unidades de Trabajo. Por ejemplo, la Unidad “Gestión de Procesos en Unix” podría tener

asignada una capacidad básica de “Administrador de procesos Unix” que formaría parte de una Unidad de Competencia “Administrador de Sistemas Unix”. Dicha capacidad estaría relacionada con otras capacidades basadas en el conocimiento de conceptos avanzados de gestión de procesos y el uso básico de un sistema operativo como Linux. En la Figura 3 se muestra esta relación entre capacidades y las unidades de trabajo asociadas a cada capacidad.

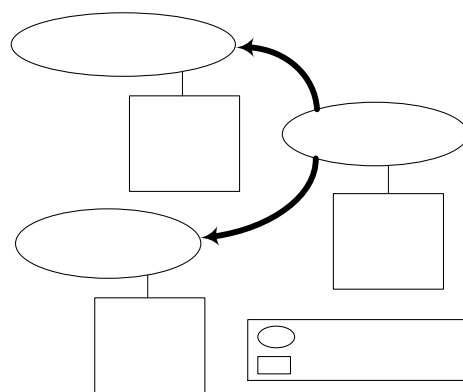


Figura 3. Ejemplo de relación entre capacidades

El modelo propuesto pretende eliminar la separación habitual entre los programas de teoría, formados por temas que estructuran los contenidos, y los programas de prácticas que incluyen las actividades a realizar en el contexto de la asignatura. La Unidad de Trabajo, como elemento básico del modelo aquí presentado, permite relacionar los apartados de teoría y práctica mediante un determinado método. Asimismo, su vinculación con otros elementos como las Unidades de Competencia y Capacidades Terminales, facilita la comprensión por parte del alumno de los objetivos de la Unidad de Trabajo.

En la Figura 4 se muestra un ejemplo global de DBT para una hipotética titulación de “Informática de Sistemas” donde se considera una Unidad de Competencia “Administrador de Sistemas Unix” asociada a un conjunto de Módulos (p.e. “Sistemas Informáticos Multiusuario”) que a su vez integran una serie de Unidades de Trabajo (p.e. “Gestión de Procesos UNIX”).

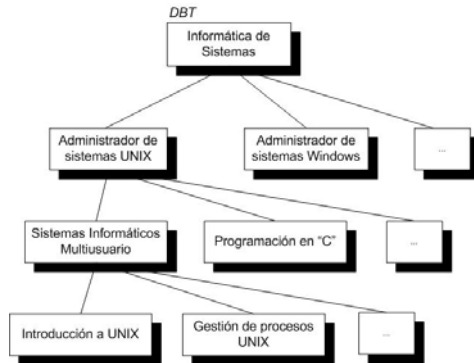


Figura 4. Ejemplo de DBT

La aplicación de este modelo a titulaciones universitarias debe dar respuesta a cuestiones como el uso del concepto de asignatura como elemento organizativo de la docencia o la realización del modelo en la impartición de determinadas disciplinas. Por un lado, las asignaturas suelen representar unidades estancas con escasa conexión con otras asignaturas. El uso de entidades como los Módulos, utilizados en Ciclos Formativos, favorecería dicha conexión y la transversalidad entre asignaturas con competencias relacionadas. Por otro lado, la realización del modelo propuesto tropieza con la división clásica de la docencia de una asignatura entre teoría y práctica. A continuación, se muestra un ejemplo práctico de implementación del modelo que resuelve dicha división en el área de Sistemas Operativos.

3. Aplicación a la enseñanza de Sistemas Operativos

Los Sistemas Operativos constituyen una disciplina básica en las titulaciones universitarias de Informática [10]. Sin embargo, no existe en el ámbito universitario una definición detallada de competencias relacionadas con dicha disciplina. En la sección 2 se ha hecho referencia a ejemplos de competencias en el uso y administración de sistemas operativos que formarían parte de un DBT en una titulación de Informática. La obtención de dichas competencias se gestionaría mediante Unidades de Trabajo de las que ahora se ofrecen algunos ejemplos. En la Figura 5 se muestra parte de una Unidad de Trabajo

dedicada a la “Gestión de Procesos POSIX”. Dicha Unidad está compuesta por una serie de elementos como:

- **Objetivos:** indican la finalidad de la Unidad de Trabajo, incluyendo las posibles capacidades a adquirir.
- **Antecedentes:** establecen los conocimientos y capacidades previas a la realización de la Unidad de Trabajo actual. En este caso se indican conocimientos de una asignatura de sistemas operativos donde se imparten fundamentos básicos de gestión de procesos. También se incluye un ejercicio para demostrar dichos conocimientos.
- **Contenidos:** indican las referencias a conceptos y procedimientos que forman parte de la tabla expuesta en la Figura 2. No se trata de la información completa sobre dichos conceptos y procedimientos sino de los puntos de acceso a esta información.
- **Actividades:** proporcionan los enlaces a los enunciados de ejercicios o tareas a resolver en el contexto de los contenidos antes presentados. Por ejemplo, un ejercicio donde se revisen las llamadas relacionadas con la creación y ejecución de procesos.
- **Evaluación:** indican los elementos como cuestionarios, ejercicios, problemas o actividades para demostrar la obtención de la capacidad o capacidades objeto de la Unidad.

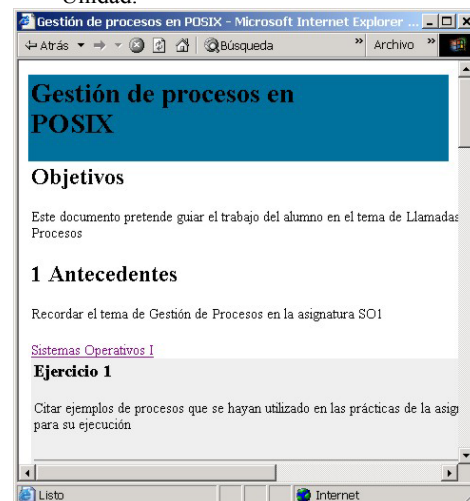


Figura 5. Representación de una Unidad de Trabajo

Algunos aspectos relevantes de este ejemplo de Unidad de Trabajo aplicada a la enseñanza de sistemas operativos son los siguientes:

- El apartado de Contenidos, está formado por referencias a conceptos o esquemas que los organizan. Tan sólo se indican definiciones u observaciones sobre conceptos básicos (por ejemplo, “la definición de proceso”), dejando al alumno la labor de búsqueda de información en las fuentes bibliográficas que se indiquen.
- Las Actividades representan el enlace entre los elementos teóricos y los prácticos de forma que se evita el aislamiento entre ambos tipos de elementos. En la Figura 6 se muestra parte del enunciado de una actividad consistente en diseñar un intérprete de órdenes sencillo [13]. Dicha actividad hace referencia también a los contenidos de gestión de procesos mencionados en la Unidad de Trabajo.
- La Evaluación no tiene porqué ser una parte aislada sino que puede estar integrada con otros apartados referidos a Contenidos y Actividades. De esta forma se permite una evaluación continua del alumno.

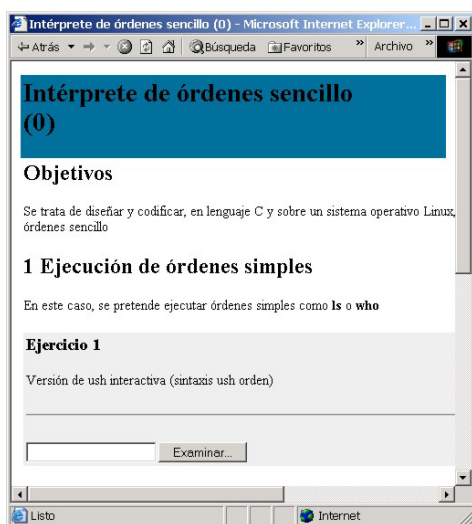


Figura 6. Representación de una Actividad

El formato empleado para visualizar los contenidos y actividades de la Unidad de Trabajo consiste en páginas HTML que se generan dinámicamente a partir de una

especificación XML de dicha Unidad de Trabajo. De esta forma se facilita la actualización de sus componentes y la publicación en diversos formatos. Las páginas HTML generadas pueden ser accedidas desde un entorno de campus virtual denominado Xedu [14].

4. Conclusiones

Este trabajo ha presentado una propuesta de modelo de diseño curricular orientado a la obtención de competencias. Una de las aportaciones ha consistido en proponer un formato de diseño curricular, ya utilizado en los Ciclos Formativos, a la enseñanza de Informática en titulaciones universitarias. Ello no significa que los estudios de los Ciclos Formativos sean comparables o del mismo nivel que los estudios universitarios sino que puede ser útil trasladar sus métodos y experiencias a este ámbito. El análisis realizado también ha servido para constatar las carencias en la definición de competencias de los informáticos a nivel universitario.

El modelo propuesto se ha basado en elementos como las Unidades de Trabajo, las Competencias a adquirir y las Capacidades Terminales que conducen a ellas como consecuencia de haber llevado a cabo las correspondientes Unidades de Trabajo. También se ha descrito una aplicación del modelo presentado a la enseñanza de Sistemas Operativos y se han revisado ejemplos de contenidos y actividades que forman parte de dicha aplicación.

Una de las ventajas que se ha observado en la aplicación del modelo consiste en la integración de elementos teóricos y prácticos mediante el uso de Unidades de Trabajo. También se ha valorado la asociación de competencias a dichas Unidades con el fin de justificar su utilidad y establecer una aproximación a las atribuciones que puede tener un profesional de la informática.

Como parte de los trabajos futuros se pretende, por un lado, analizar a corto plazo la implementación del modelo en el ámbito de un grupo de enseñanza no presencial en una asignatura de Sistemas Operativos. Este grupo forma parte de una iniciativa de tele-enseñanza

de la Escuela Técnica Superior de Informática Aplicada en la Universidad Politécnica de Valencia. Por otro lado, se trata de analizar en próximas comunicaciones la utilidad del modelo aportado de cara a la futura integración de las titulaciones informáticas a nivel europeo.

Referencias

- [1] Ley Orgánica de la Universidad, obtenido el 30/01/2004 de la página web: <http://www.univ.mecd.es/univ/jsp/plantilla.jsp?id=231>
- [2] Declaración de Bolonia, obtenido el 30/01/2004 de la página web: <http://www.universia.es/contenidos3/internacionales/Bolonia/indice.htm>
- [3] Sarramona, J., "Principios básicos de diseño curricular". Ponencia Proyecto de Innovación Educativa. I.C.E., Universidad Politécnica de Valencia. 1988.
- [4] Escudero, D. "Proyecto Docente", obtenido el 30/01/2004 de la página web: <http://www.infor.uva.es/~descuder/docencia/pd/pd.html>
- [5] Plan de Estudios de Ingeniería Técnica Informática de Sistemas de la Universidad Politécnica de Valencia (UPV), obtenido el 30/01/2004 de la página web: http://www.eui.upv.es/webei/la_escuela/titulaciones/ITIS01/ITIS01.php
- [6] Modelos curriculares para Informática y Computación, obtenido el 30/01/2004 de la página web: <http://www.somece.org.mx/memorias/1999/docs/ponen79.doc>
- [7] Directrices curriculares de cursos del área de Computación e Informática, obtenido el 30/01/2004 de la página web: <http://www.inf.ufgrs.br/mec/ceeinf.diretrizes.html>
- [8] Plan de Estudios de Ingeniería Informática Universidad Autónoma de Madrid (UAM), obtenido el 30/01/2004 de la página web: <http://www.uam.es/estudiantes/acceso/Estudios/Licenciaturas/pdf/Informatica.pdf>
- [9] Familia Profesional de Grado Superior de Informática. Administración de Sistemas Informáticos, obtenido el 30/01/2004 de la página web: http://www.mec.es/fp/catalogos/gsinfor_admSis.shtml
- [10] ACM/IEEE Task Force on the Year 2001 Model Curricula for Computing, Computing Curricula 2001(CC-2001), obtenido el 10/11/1999 de la página web: <http://www.computer.org/education/cc2001/>
- [11] Propuesta de diseño de titulación y plan de estudios de Biblioteconomía y Documentación, obtenido el 30/01/2004 de la página web: <http://www.ub.es/biblio/convergencia-europea-propuesta.html>
- [12] CEFIRE Formación Profesional, obtenido el 30/01/2004 de la página web: <http://intercentres.cult.gva.es/cefire/46402871/>
- [13] Carretero, J., García, F., Perez, F. Prácticas de Sistemas Operativos. Ed. McGraw Hill.
- [14] Buendía, F., Agustí, M., Benlloch, J.V., Bisbal, E., Lluésma, M. XEDU, an Open Learning Management System Proposal. International Conference on Network Universities and E-learning'03 Valencia, Mayo, 2003.