

MEMORIA DESCRIPTIVA DE ACCIONES DE INNOVACIÓN DOCENTE

UNIVERSIDAD DE GRANADA

❖ DATOS IDENTIFICATIVOS:

Título de la acción

Un Entorno para el Aprendizaje en Sistemas Operativos I

Resumen de la acción

El proyecto ha concluido con la consecución de los objetivos propuestos y la generación del siguiente material para las asignaturas de Sistemas Operativos I que se imparten en la E.T.S. de Ingeniería Informática:

1. Creación de material docente destinado a mejorar la calidad docente de la asignatura. Este material, que difiere en su contenido y formato, puede clasificarse como:
 - a. Animaciones de ordenador que muestran el comportamiento de determinados aspectos de funcionamiento del sistema operativo.
 - b. Presentaciones que permiten mostrar el comportamiento dinámico de aquellos aspectos difíciles de mostrar a través de las animaciones de ordenador. Básicamente la diferencia con las anteriores es que su dificultad requiere que las animaciones se deba realizar manualmente.
 - c. Tutoriales que amplían los materiales existentes de la asignatura y permiten mostrar la transversalidad de determinados contenidos de la asignatura.
 - d. Realización de programas de ordenador destinados a mostrar determinados aspectos vistos en clase.
 - e. FAQ's destinadas a contestar a las cuestiones o dudas más frecuentemente realizadas por los alumnos respecto a las prácticas de las asignaturas.
2. Creación de una página web que centraliza toda la información relevante de las asignaturas de Sistemas Operativos, pero a la vez permite especializar determinados contenidos (listas, calificaciones, etc.) en función de la titulación y los grupos.
3. Recopilación de software destinado a facilitar la realización de las prácticas de programación con hebras.
4. Recopilación y selección de enlaces a páginas web de recursos útiles relacionados con la disciplina.

Componentes del grupo

Nombre y apellidos	Área de Conocimiento	Departamento
--------------------	----------------------	--------------

Coordinador/a: José Antonio Gómez Hernández - Lenguajes y Sistemas Informáticos

Componentes: Alejandro León Salas	“
Paderewski Rodríguez, Patricia	“
Ruíz Martín, Emilia	“
Sánchez Buendía, M ^a Angustias	“

Ámbito de actuación de la acción

Área de conocimiento Departamento Titulación Centro

Asignaturas afectadas

Nombre de la asignatura	Área de Conocimiento	Titulación/es
Sistemas Operativos I	Lenguajes y Sistemas Informáticos	Ing. Informática
Sistemas Operativos I	Lenguajes y Sistemas Informáticos	Ing. Tec. Inf. Sistemas
Sistemas Operativos I	Lenguajes y Sistemas Informáticos	Ing. Tec. Inf. Gestión

MEMORIA DE LA ACCIÓN

- La Memoria debe contener un mínimo de 10 páginas y un máximo de 15 páginas y debe contener los apartados señalados.. Escriba el texto dentro de los recuadros correspondientes.
- En el caso de que durante el desarrollo de la acción se hubieran producido documentos o material dignos de reseñar (CD, páginas web, revistas, vídeos, etc.) se aconseja incluir como anexo una copia de buena calidad de los mismos a efectos de evaluación.

1. Introducción (Justificación del trabajo, contexto, experiencias previas...)

Vamos a exponer las razones que nos han llevado a establecer los objetivos del proyecto y que acabamos de mencionar. Pero antes, realizaremos una breve descripción de las asignaturas y valoraremos los resultados académicos de algunos cursos para poder centrar el problema con que nos encontramos los profesores de las mismas.

Las asignaturas de Sistemas Operativos I son troncales de segundo curso de las tres titulaciones de Informática que se imparten en la Universidad de Granada. Las tres asignaturas tienen 4.5 créditos de teoría y 1.5 de prácticas, lo que supone un total de 6 créditos (5.33 créditos ECTS), es decir, 4 horas semanales cuatrimestrales (tres horas de teoría y 1 hora de prácticas) de docencia presencial.

Las asignaturas recogen los principales temas de sistemas operativos que aparecen en los currícula de Informática como los últimos propuestos por la ACM/IEEE de 2001[1]. En concreto, el temario de la asignatura recoge los siguientes puntos: Introducción a los sistemas operativos, Procesos y hebras, Sincronización y comunicación entre procesos, Gestión de memoria, y Sistemas de archivos. Los restantes temas recogidos en los currícula se estudian en las asignaturas de Sistemas Operativos II que se imparten en el segundo cuatrimestre del mismo curso. Estas asignaturas son obligatorias de universidad y tienen los mismos créditos que las anteriores.

El programa de teoría de las asignaturas es un “programa clásico”. Si bien, como se justifica en [2], se han ajustado los contenidos respecto a los textos clásicos de la disciplina [3, 4, y 5] para eliminar aquellos elementos que han quedado obsoletos, y por tanto no son de aplicación, permitiendo que se dedique más tiempo al estudio de conceptos de validez actual.

En cuanto a las prácticas de las asignaturas introductorias de sistemas operativos existen dos grandes enfoques utilizados tanto a nivel nacional como internacional:

- 1) Uso de simuladores para el desarrollo de los conceptos vistos en clase de teoría [6].
- 2) Uso de los servicios del sistema operativo como usuario avanzado, tanto desde la interfaz suministrada por el intérprete de órdenes, o shell, como desde la interfaz suministrada por las llamadas al sistema y bibliotecas especializadas.

En nuestro caso, hemos optado por el segundo enfoque por las siguientes razones:

- a) Si bien la utilización de simuladores de hardware o de sistemas operativos –primer enfoque- permite una comprensión exhaustiva de los conceptos de sistemas operativos, utilizada en una asignatura introductoria de la materia, presenta el problema de introducir una complejidad innecesaria y, lo que es peor, sólo permite cubrir algunos de los conceptos vistos en clase. El alumno puede comprender bien un concepto pero como dedica mucho tiempo a su implementación esto le impide abordar otros conceptos igualmente importantes. Nuestra experiencia en este terreno se recoge en [7].
- b) Nuestros estudiantes cuando llegan a segundo solo tienen un conocimiento rudimentario de cómo funciona un sistema operativo. En este sentido, pretendemos que la asignatura le permita conocer no sólo como se construye un sistema operativo sino también como obtener de él servicios avanzados. Este conocimiento será muy útil para el desarrollo eficiente de las prácticas de otras asignaturas como en su carrera profesional.
- c) El uso combinado del estudio de las interfaces del sistema operativo les da una perspectiva enriquecedora sobre qué servicios se pueden obtener del sistema y en qué condiciones y cómo se implementan estos. Esto es útil no solo para su conocimiento en la materia, sino que además se extraen conocimientos que son aplicables al desarrollo de software en general.

Dado que nuestros alumnos cuando llegan a segundo han visto muy poco del sistema operativo Unix, las prácticas se dividen en dos partes diferenciadas pero interrelacionadas. En un primer bloque, lo que hacemos es introducirlos en un shell de Linux hasta que adquieran un nivel suficiente para desarrollar programas shells básicos que le serán útiles en su trabajo diario tanto en los estudios como en su actividad profesional. En un segundo bloque se les introduce a la programación de sistemas utilizando la biblioteca de hebras que permite un doble objetivo, ilustrar los conceptos de sincronización y comunicación entre hebras vistos en clase, y disponer de una herramienta de programación muy utilizada en aplicaciones reales. Este segundo bloque luego se amplía en Sistemas Operativos II con el estudio de las principales llamadas al sistema operativo.

El enfoque de diseño de componentes reales de sistemas operativos se completa en la asignatura optativa de Diseño de Sistemas Operativos de 5º curso, una vez que los alumnos han cursado las asignaturas básicas de sistemas operativos así como otras asignaturas necesarias como son las relacionadas con estructura y arquitecturas de Computadores. En esta asignatura se implementan servicios de un sistema operativo real (Linux) no de uno simulado.

Respecto a los resultados académicos, las estadísticas nos indican (ver Memoria Extendida) que la suma de los que no superan la asignatura y los no presentados ronda el 50%.

Al sondear a los estudiantes sobre las dificultades de las asignaturas hay un acuerdo unánime de que el principal escollo se encuentra en la abstracción de los conceptos que se estudian. Estos datos han impulsado en los profesores de la asignatura dos tipos de medias a tomar:

- 1) La realización de este proyecto de innovación encaminado al desarrollo de material docente que facilite el estudio de los conceptos más difíciles de entender de las asignaturas.
- 2) Otro aspecto metodológico, encaminado a utilizar el aprendizaje cooperativo como medio de mejorar la calidad de la docencia. En este sentido, parte de los profesores del proyecto estamos involucrados en un grupo de trabajo de los Proyectos de Formación del Profesorado patrocinados por la UCUA.

Como acabamos de indicar, el estudiante encuentra dificultades en la comprensión de determinados conceptos de las asignaturas. Por ello, vamos a citar los principales escollos que suelen encontrar nuestros alumnos detectados durante la impartición de la citada materia. La exposición de los puntos concretos la vamos a estructurar según si se encuentran en teoría o en prácticas.

A - Puntos difíciles de teoría

Los dos principales temas donde encontramos las dificultades son precisamente en el núcleo principal de las asignaturas, los temas de relacionados con la concurrencia. A saber:

- (a) Tema 2 dedicado a la implementación de procesos y hebras
- (b) Tema 3 dedicado a la sincronización y comunicación entre procesos/hebras.

En el tema 2, nos encontramos problemas a la hora de estudiar la abstracción proceso realizada por el sistema operativo. La mayoría de los estudiantes tiene la idea de que los procesos son algo dado por la propia máquina y cuesta hacerles ver que no son algo que existe en la máquina sino que son construidos por el sistema operativo. En cuanto a la parte de hebras, por una parte, uno de los problemas radica en la novedad del concepto, otro está íntimamente ligado con el anterior de creación de un proceso. En concreto, los conceptos más complicados son: implementación de la multiprogramación, reentrada y apropiatividad.

El Tema 3 presenta un aspecto totalmente nuevo para ellos, la programación concurrente. Acostumbrados desde sus primeros contactos con la programación secuencial, tienen serias dificultades para acostumbrarse a plantear soluciones concurrentes a los problemas de programación que encontramos normalmente. En concreto, los principales puntos que presentan dificultades en este tema son:

- Ser conscientes del problema de la condición de carrera
- Detectar las secciones críticas y por tanto en las que hay que aplicar exclusión mutua.
- Resolución concurrente de problemas básicos.

En el resto de temas, las principales dudas surgen como consecuencia de la aparente simplicidad de los conceptos teóricos tal como se exponen en la bibliografía. La solución a estos problemas radica básicamente en hacerles ver que el esquema mental que producen al estudiar estos conceptos no es el correcto. Por ejemplo, encontramos con frecuencia que nuestros estudiantes cuando estudian el mecanismo de traducción de direcciones mediante la paginación son capaces de repetir el proceso teórico pero en muchos casos son incapaces de resolver un problema concreto donde haya que traducir direcciones de un ejemplo concreto. Para solucionar este tipo de problemas, basta en general, proponer y resolver los problemas adecuados.

B- Puntos difíciles de prácticas

Como indicábamos en el apartado de descripción de las asignaturas, las prácticas están divididas en dos grandes bloques:

- (a) Manejo de un shell y órdenes básicas de Unix
- (b) Programación concurrente con la biblioteca de hebras de Linux

Respecto al primer bloque, la principal dificultad radica en que actualmente el contacto de nuestro estudiantes con el sistema operativo, tanto Windows como Linux, es casi exclusivamente a través de una interfaz gráfica. Esto dificulta el correcto uso de una interfaz de texto que es menos agradable y más complicada, pero también más potente. En este sentido, es difícil que los estudiantes comprendan y manejen Linux que esta basado en una filosofía de "caja de herramientas". Les cuesta construir una orden compleja basándose en órdenes más sencillas y elementales.

En el segundo bloque, podemos encontrar además de los problemas citados para el Tema 3 de teoría, es decir, de los derivados de la programación concurrente, y también los ocasionados por no haber utilizado C, sino C++ en cursos anteriores.

En ambos bloques, nos encontramos el problema de que nuestros estudiantes tienen dificultades a la hora de comprender los mensajes de error, dados por el shell o el compilador, y lo que es peor aún, de desarrollar estrategias para localizar los errores y repararlos.

2. Objetivos (Concretar qué se pretendió con la experiencia)

Como indicábamos en el apartado anterior, el objetivo básico del proyecto ha sido desarrollar o centralizar material docente en diferentes formatos (simulaciones, software, presentaciones, etc.) que permita al estudiante una mejor comprensión de los conceptos estudiados. Este material tiene un doble uso. Por una parte, puede en su mayoría utilizarse en clase, de otro lado, puede utilizarse como material de autoaprendizaje para que el estudiante acceda a él desde casa.

Los objetivos concretos del proyecto son los siguientes:

- 1) Utilizar diferentes técnicas (animación, programas, etc.) para ilustrar los contenidos de la asignatura Sistemas Operativos I que son más difíciles de comprender por parte de los alumnos. En especial, los contenidos derivados del comportamiento dinámico del sistema operativo. Estas técnicas podrán ser utilizadas tanto en la docencia presencial como a través de Internet, lo que permitirá que el alumno las trabaje con más detenimiento desde casa, la biblioteca, etc.
- 2) Desarrollar herramientas, así como una guía, de ayuda a la resolución de los problemas más frecuentemente encontrados por los estudiantes en la realización de los ejercicios y las prácticas de las asignaturas.
- 3) Centralizar en una única página web los recursos propios desarrollados como producto de este proyecto junto a los desarrollados por otras universidades y que podemos encontrar en Internet.
- 4) En la línea anterior, centralizar los contenidos en una página que servirá para unificar las distintas páginas de las asignaturas de Sistemas Operativos I que mantenemos los profesores de las asignaturas, ya que permitirá poner en la página los elementos comunes a las asignaturas (objetivos, temario, bibliografía, etc.). También se permitirá personalizar algunas partes que diferencien las diferentes titulaciones y/o grupos de las mismas como pueden ser listas de calificaciones, asignación de trabajos, etc.

3.Descripción de la experiencia (Exponer con suficiente detalle lo realizado en la experiencia)

Los profesores que participamos en este proyecto llevamos muchos años impartiendo diferentes asignaturas de sistemas operativos y en especial las de Sistemas Operativos I en las tres titulaciones que existen en la actualidad en los estudios de Informática en nuestra Universidad. Esto nos ha permitido detectar aquellos conceptos, ideas o aplicaciones que suelen tener mayor dificultad de asimilación por parte de los alumnos de las citadas asignaturas, tanto en teoría como en prácticas. En este sentido, el objetivo del proyecto ha sido elaborar material concreto, tanto para teoría como para prácticas, que facilite la labor de comprensión de aquellos aspectos previamente detectados como los de más difícil comprensión por el alumno.

Desde nuestro punto de vista la asignatura de sistemas operativos, troncal de los planes de estudios, tiene una doble importancia:

1. Suministra una cultura general sobre qué es, para que sirve, y cómo se implementa un sistema operativo
2. El estudiante debe conocer los efectos prácticos del funcionamiento del sistema operativo dado que éste será parte de la mayoría de los sistemas de computación que el alumno maneje y, por tanto, interaccionará con el constantemente (directamente o a través de sus aplicaciones). Además, el sistema operativo no es más que un sistema software complejo y todos los problemas de su diseño y desarrollo son aplicables a otros sistemas software generales.

Las asignaturas introductorias de sistemas operativos suelen tener, desde nuestro punto de vista, un problema. Los cursos de introducción, y así lo recoge la bibliografía de la disciplina, suelen estar pensados para dar a conocer al estudiante las principales abstracciones utilizadas en SOs y cómo se pueden implementar en la mayoría de los casos. Esto así expuesto parece razonable. El problema surge cuando en el desarrollo de la labor docente se comprueba que estas descripciones generales no son suficientes para comprender y manejar correctamente un concepto. En estos casos, es necesario bajar a nivel de implementación para aclarar la realidad del concepto. Aquí, nos aparece un problema difícil de remediar, la aparente sencillez de los conceptos en estado abstracto se ve rota por la dificultad de la implementación debida a la multitud de detalles que nos aparecen a nivel de dicha implementación. Es este sentido, el material desarrollado intenta llegar a un equilibrio entre el concepto a explicar y el nivel de descenso necesario.

El descenso a nivel de implementación es importante desde nuestra experiencia pues conecta al estudiante con la realidad de un sistema operativo y por tanto lo obliga a relacionar los conceptos vistos en clase con las dificultades que puede encontrar en el uso diario de un sistema operativo.

Algunos de los conceptos concretos en los que se encuentran más dificultades y que han sido objetos de estudio en este proyecto, tenemos:

a) Teoría

- Abstracción proceso y su vertiente actual de hebras.
- Implementación de la operación cambio de contexto.
- Implementación de modo dual de funcionamiento.
- Temas relacionados con la sincronización de actividades entre procesos y hebras
- Implementación de memoria en sistemas paginados y segmentados
- Entender el aspecto dinámico de determinadas facetas de un sistema operativo.

b) Prácticas

- Depuración de programas multihebrados
- El uso de entornos gráficos muy utilizado actualmente por todos dificulta a los estudiantes comprender y usar con facilidad un entorno de órdenes textual como un shell no gráfico. En este sentido, los estudiantes tienen dificultades para entender los mensajes de error dados por las herramientas, y por tanto, esto acarrea dificultades para poder remediarlos.

Para resolver estas dificultades se han realizado las siguientes propuestas. Algunos de los métodos propuestos en el proyecto han sido utilizados con anterioridad si bien se han mejorado en cuanto a los contenidos y presentación. Estos métodos se han mostrado como una gran ayuda a la hora de comprender los conceptos que ilustran. Otros métodos son nuevos y han sido expresamente realizados para este proyecto.

En concreto podemos agrupar los métodos usados en cuatro categorías:

- a) Uso de simuladores para la animación de conceptos de sistemas operativos – la visualización de los algoritmos utilizados en sistemas operativos suele ser tediosa de realizar en pizarra o presentación por lo que se recurre a la animación software que son bastante potentes y simples de realizar.
- b) Desarrollo de presentaciones y material escrito que extiende y aclara el material que existe en la bibliografía de la disciplina – Este material se utiliza para el estudio de conceptos difíciles de comprender y que no son fácilmente representables gráficamente por lo que no se pueden utilizar animaciones gráficas.
- c) Uso de software “real” como apoyo a la docencia – en este grupo entra la utilización de software de depuración utilizado en el desarrollo de programas pero que se puede utilizar con fines docentes para detectar con facilidad los problemas que encuentran los alumnos en la realización de las prácticas. También, demostración de algunos conceptos vistos en clase utilizando el portátil que sirve para exponer las transparencias con varios sistemas operativos (Windows y Linux).
- d) Recopilación de enlaces de interés que permitan al estudiante profundizar en los temas, ver otras perspectivas de planteamiento y solución de problemas, etc. Esta recopilación se presenta como colección de enlaces web que permiten al alumno acceder al material indicado de una manera fácil y desde cualquier lugar.

Además, el material desarrollado o utilizado en este proyecto amplía o repite el material presentado ya en las transparencias de clase, pero desde otro enfoque. Esta repetición permite ver el problema desde otra perspectiva lo que es de gran interés, en especial pensándolo como material de autoformación por el siguiente motivo. En ocasiones hay diferentes visiones, ejemplos o formas de exponer un concepto. El profesor elige una de ellas para usarla en clase utilizando diversos criterios: es la más fácil, puede ser comprendido por la mayoría, que conocimientos previos tienen, etc. Esta forma elegida puede no ser la más adecuada para algunos alumnos. Por ello, la exposición adicional de nuevo material que explique el mismo concepto desde diferentes puntos de vista puede permitir su visualización por parte de aquellos alumnos que no lo han visto con el primer enfoque. Además, para el resto también puede ser enriquecedor pues puede hacerles ver nuevas relaciones.

Otro aspecto que pretende abordar el proyecto es el de la transversalidad de contenidos de la propia asignatura y de esta en su relación con otras. La linealidad de la exposición de clase hace que en ocasiones se pierda la transversalidad entre contenidos de la asignatura. Así mismo la estructuración de las disciplinas en asignaturas hace que en ocasiones se pierda la transversalidad de contenidos que están en los límites de contacto de las asignaturas. En ocasiones, los contenidos pueden verse varias veces de diferente manera lo que hacen que parezcan diferentes, otras veces simplemente no se ven pensado que el otro profesor los ha cubierto.

Este proyecto también pretende recoger parte de esta transversalidad. En este sentido, parte del material desarrollado “duplica” los contenidos vistos en clase al objeto de poder relacionarlos con otras asignaturas.

El ejemplo típico de esta interacción entre asignaturas es la gestión de memoria en un sistema operativo. Este servicio del sistema operativo lo podemos ver como un banco con tres patas, una de ellas los sistemas operativos, otra la arquitectura hardware de la máquina (en concreto, cachés y MMU) y la última la generación de código por parte del compilador.

En cuanto a la integración del material desarrollado, se ha optado por integrarlo todo en una página web, lo que permite tanto su utilización en clase como material de apoyo a las aplicaciones como su uso por parte del alumno individualmente como material de autoformación.

El material concreto desarrollado se muestra en los diferentes apéndices de la memoria agrupados por las categorías antes indicadas. La estructura de cada uno de los apéndices es la siguiente: primero, se expone el problema que se pretende solventar; segundo, se presenta el material desarrollado.

4. Material y métodos (Describir la metodología seguida y, en su caso, el material utilizado)

▪ Metodología

Los principales pasos seguidos en la realización del presente proyecto ha sido los siguientes:

- 1) Identificar aquellos conceptos o contenidos con mayores dificultades de comprensión por parte de los estudiantes.
- 2) Cribar todos estos puntos difíciles para trabajar sobre aquellos que una primera instancia son más relevantes para la obtención de un nivel de conocimiento correcto de nuestros alumnos.
- 3) Determinar el material a desarrollar para mejorar el aprendizaje de los conceptos básicos de la disciplina.
- 4) Analizar la calidad del material usado y su adecuación a los objetivos propuestos.

Para el punto 1, hemos partido de la amplia experiencia docente de todos los profesores que componemos el grupo de trabajo, lo que nos ha permitido generar una lista de todos aquellos conceptos que presentan dificultades de aprendizaje.

En la fase 2, una serie de reuniones de grupo sirvieron para ir descartando de la lista inicial los elementos de segundo orden, y quedarnos solo con los básicos para un buen conocimiento de la materia.

El tercer punto, la elección del material que debía ilustrar los conceptos complejos, se realizó a su vez en tres fases:

- 1º Se localizó todo el material existente, a nivel nacional como internacional, desarrollado para la docencia de sistemas operativos.
- 2º De todo el material localizado, se vio cual era útil para nuestras asignaturas y cual no encajaba en nuestros programas o dentro de nuestros objetivos. El material externo que es de utilidad lo hemos hecho accesible a nuestros alumnos a través de enlaces web para que puedan consultarlo, ya que nos parecía un desperdicio de tiempo replicar ese material.
- 3º Para aquellos elementos para los que no se encontró material o no era de la suficiente calidad, se generó el mismo. Dependiendo del objeto de estudio se optó por generar un tipo de material u otro. En unos casos han sido programas ejemplo de ordenador, en otros animaciones para ilustrar los caminos de código dentro del sistema operativo o las acciones que tiene lugar en el sistema. También, hemos recopilado las preguntas más frecuentes de los estudiantes para acelerar la solución de problemas sencillos, o hemos resuelto ejercicios u problemas como muestra de cual es el proceso que ellos deben seguir a la hora de abordar los problemas de clase o los que se les puedan presentar.

Respecto al análisis de los resultados, fase 4, pretendemos estudiarlos en el próximo curso donde ya tendremos disponibles para los alumnos todos los materiales a través de una página web en los servidores del departamento y se aplicarán de forma sistemática. Si bien, los datos que tenemos hasta la fecha, resultado de los exámenes, preguntas de los alumnos en clase, y charlas con ellos en tutorías, son prometedores.

▪ Material utilizado

Se han utilizado para el desarrollo del proyecto los PCs que tenemos disponibles en el departamento (los personales y los servidores para alojar las páginas webs), las impresoras, así como diverso material fungible (papel, discos, etc.).

Así mismo con el proyecto se han adquirido dos portátiles para poder utilizar el material desarrollado en las clases presenciales. Y se va adquirir soporte, aún en estudio, para mantener el material de clase siempre accesible y no depender de la impredecibilidad de la red, que si bien esta disponible gran parte del tiempo hay cortes en los momentos más inoportunos. Este material puede ir desde memorias flash para mantener todas las páginas web, hasta una grabadora y disco para tenerlo accesible en CD-ROM.

5.Resultados obtenidos y disponibilidad de uso (Concretar y discutir los resultados obtenidos y aquellos no logrados, incluyendo el material elaborado y su grado de disponibilidad).

Agrupado por categorías, la relación de material elaborado para el proyecto es:

- **Animaciones** de algoritmos de gestión de recursos (CPU, discos, y memoria) que permiten estudiar, y probar los principales algoritmos que utiliza un sistema operativo para gestionar sus recursos. Entre estas animaciones, hay programas desarrollados en otras universidades y programas desarrollados por nosotros. Para su visualización en la mayoría de los casos sólo es necesario un navegador y visor de applets. Solo en el caso de VisualOS es necesario instalar el software en un sistema Linux, teniendo la ventaja que es una herramienta de dominio público por lo que es accesible por los estudiantes a coste cero.
- **Presentaciones** Power-Point que ilustran aspectos dinámicos del funcionamiento del sistema más complejos que la simple animación de un algoritmo. Estas presentaciones pretenden que el estudiante visualice paso a paso los caminos de código involucrados en determinados aspectos del funcionamiento del sistema operativos. Por ejemplo, los contenidos de algunas presentaciones son: realización de una llamada al sistema, cambio de contexto, implementación de la apropiatividad y la no apropiatividad de un algoritmo de planificación de la CPU, etc. Para poder verlas solo es necesario un visor de presentaciones ya sea MS Power-Point u OpenOffice.
- **Programas ejemplos** implementados para ilustrar ciertos problemas planteados en clase y sus soluciones. Estos programas hacen tangibles ciertos problemas que por mucho que se explique de forma teórica no son entendidos como tales pues no se le han presentado aún al alumno en su trabajo diario. A su vez, se implementan algunas soluciones para estos problemas. El estudiante puede descargar los programas y jugar con ellos: ver el código, modificarlo, implementar nuevas soluciones, etc. Para llevar a cabo estas acciones el estudiante solo necesita un compilador de C. En este apartado se han implementado programas para visualizar la condición de carrera que se plantea cuando varios procesos/hebras comparten variables comunes, también se han implementado soluciones para evitar las condiciones de carrera a través del algoritmo de Lamport, de semáforos, etc. Se ha resuelto los problemas de productor-consumidor, de los lectores-escritores, y los filósofos comensales (aquí se hace visible el interbloqueo). Por último, un programa ejemplo ilustra las operaciones básicas sobre procesos y otro el concepto de entradas/salidas asíncronas.
- **Tutoriales** o material textual complementario que profundiza en contenidos vistos en clase de forma básica y que están a caballo entre varias asignaturas. En este sentido, los tutoriales “Programación del x86” y “Compilación de programas” permite mostrar con detalle la realidad que esconde el interior del sistema operativo, así como ver las relaciones con otras disciplinas como son la Arquitectura de Computadores y los Compiladores. Un tutorial sobre “Reentrancia” profundiza en este aspecto importante a la hora de construir programas multihebrados que les es útil para las prácticas de esta y otras asignaturas, y sobre todo para su futuro profesional.
- **Sermones** o breves disertaciones para llamar de forma rápida la atención del estudiante sobre aspectos importantes de su labor tanto técnicos como éticos.
- **Software** “real” utilizado para una mejor y más rápida solución a las prácticas planteadas en clase. En este sentido, el depurador Valgrind [9] es una importante ayuda para la depuración de las aplicaciones multihebradas que se les proponen en clase. Este depurador es de dominio público.
- **Soluciones de ejercicios** que le permiten ver con se resuelven algunos de los supuestos prácticos tanto de teoría como en prácticas. En este sentido, se han realizado una batería de ejercicios resueltos para el manejo del shell y las órdenes del sistema Linux, y se han solucionado con detalles los principales problemas de sincronización. Para este último caso, donde hemos detectado que los estudiantes tienen problemas a la hora de desarrollar las soluciones de problemas de sincronización, se ha explicitado paso a paso la solución de un problema de sincronización con las oportunas justificaciones que por qué se introducen los diferentes semáforos y variables.

- **FAQ:** dos FAQ, una sobre la programación en tc-shell y otra de programación con hebras, permiten al estudiante resolver de forma rápida las dudas iniciales que surgen cuando se abordan las prácticas por vez primera. Esto acelera el aprendizaje del estudiante al aclarar ciertas dudas de una forma rápida, sin necesidad de esperar a la siguiente clase de prácticas o a tutorías.
- **Colección de enlaces** – Se ha confeccionado una relación de enlaces a páginas web que permite el acceso a recursos relacionados con los sistemas operativos. Esta colección se divide en enlaces relacionados con la docencia y enlaces no docentes. En los enlaces relacionados con la docencia, podemos encontrar a su vez dos grupos: los que permite acceder a páginas web de asignaturas similares en España y fuera de nuestro país, y los que permiten acceder a los recursos docentes suministrados por los autores de los principales libros de la disciplina. En cuanto a los enlaces no docentes, encontraremos enlaces a páginas diversas que ponen al estudiante con la realidad de los que se está realizando en el mundo sobre el tema: diseño e implementación de sistemas operativos docentes, comerciales, de dominio público, y de investigación, estudios sobre los sistemas operativos más frecuentes, sus ventajas e inconvenientes, artículos de actualidad, etc.

Desde el principio tuvimos claro, que el material debía ser accesible para el estudiante fuera del aula, por ello decidimos que todo debería ser integrable en una página web para que se pueda consultar con las aplicaciones básicas de un sistema informático: sistema operativo, navegador de Internet y un visor de documentos. El material se encontrará en breve disponible en la dirección <http://www-lsi.ugr.es/~jagomez/soi.html>, hasta que se decida la ubicación definitiva.

6. Utilidad de la experiencia (Comentar para qué ha servido la experiencia y a quienes o en qué contextos podría ser útil).

La realización del presente proyecto ha sido para nosotros una experiencia enriquecedora por varios motivos.

Primero, ha servido como vehículo de comunicación los compañeros que imparten estas asignaturas en otras titulaciones o grupos y para intercambiar experiencias docentes sobre los métodos y materiales que van bien, los que no, etc.

En segundo lugar, el desarrollo sistemático de material para la asignatura ha servido sin duda para mejorar la calidad de la docencia de las mismas al darles más cuerpo y coherencia a los contenidos de las mismas. En este sentido, el simple hecho de generar material docente para ilustrar un determinado concepto hace ver a nuestros alumnos cuales son aquellos puntos que tienen una mayor importancia no solo dentro del contexto de la asignatura sino también para su futuro desarrollo profesional.

Por último, como docentes que somos, ha supuesto una satisfacción personal el ver que mejoran los conocimientos adquiridos por nuestros alumnos. No solo por que el material generado les aclara las ideas, sino que además les plantea nuevos retos fomentando su curiosidad sobre como realmente funciona un sistema operativo.

El material desarrollado es útil para cualquier asignatura introductoria de Sistemas Operativos, dentro y fuera de nuestra Universidad, puesto que ilustra los conceptos básicos de los sistemas operativos. En este sentido, se puede aplicar a las asignaturas actuales, así como a cualquier asignatura básica que pueda surgir tanto por la reforma actual de títulos como por la aparición de nuevas titulaciones relacionadas con las tecnologías de la información.

7. Observaciones y comentarios (Comentar aspectos no incluidos en los demás apartados)

Por ello, es nuestro objetivo docente seguir trabajando en esta línea para poder extender el material docente de forma que abarque otros contenidos no cubiertos por el presente proyecto.

También, extender de forma sistemática esta iniciativa a la asignatura de Sistemas Operativos II, donde como ocurría con Sistemas Operativos I se han realizado algunos trabajos pero no de forma sistemática.

En concreto hay un punto de interés que aún no está resuelto y que nos gustaría abordar. Del análisis de las diferentes animaciones de algoritmos que hay en circulación se puede observar que todas simulan algoritmos sencillos, como los de planificación de la CPU o disco, gestión de memoria, etc. Pero no existe simulación que recojan los aspectos dinámicos de funcionamiento del sistema. Esto se debe a que no existe una forma de representación simple de estos aspectos. Por ello, nuestra intención es seguir trabajando en encontrar una forma de representación visualmente sencilla de los aspectos dinámicos del sistema, por ejemplo, cómo representar el funcionamiento de un kernel no apropiativo cuando se está ejecutando una hebra y se produce una interrupción, o cuando se anidan las ejecuciones de varias interrupciones. En esta línea, creemos que es posible utilizar herramientas como LTT [9] (Linux Trace Toolkit), que si bien están diseñadas para trazar el kernel de Linux, pueden servir para analizar los caminos de código kernel y por tanto ver el comportamiento dinámico del sistema.

8. Autoevaluación de la experiencia (Señalar la metodología utilizada en la evaluación y los resultados de la experiencia)

De todo el material desarrollado en el proyecto, solo una parte pequeña ha sido puesta a prueba con anterioridad. Por ello, hasta la fecha no hemos establecido un procedimiento de evaluación que nos permita determinar el grado de calidad del material elaborado y, sobre todo, la utilidad para el estudiante de ese material.

Para el próximo curso académico, se pondrá en funcionamiento todo el material desarrollado y podremos tener cifras sobre la utilidad del material. Para recabar información, tenemos planteadas dos líneas:

1) Pasar una encuesta a nuestros estudiantes cuando finalice el cuatrimestre destinada tanto a recabar información sobre la utilidad del material desarrollado, como detectar cuales son los puntos débiles y los puntos que quedan por cubrir.

2) Como no todos los profesores que imparten docencia en las asignaturas de Sistemas Operativos I están involucrados en el proyecto de innovación, y presumiblemente no todos utilizarán el material del presente proyecto, podemos utilizar estos grupos como grupos de referencia para analizar si con el material generado se mejora el aprendizaje de los estudiantes.

No obstante, de manera informal, podemos indicar que a la vista tanto de las preguntas planteadas en clase por los estudiantes como de las respuestas a las preguntas de los exámenes finales, vemos que el estudiante maneja mejor los conceptos para los cuales se ha desarrollado material adicional, y sabe relacionarlo mejor con otros conceptos.

9. Bibliografía

- [1] IEEE/ACM, *Computing Curricula 2001 Computer Science*, The Joint Task Force on Computing Curricula. IEEE Computer Society and Association for Computing Machinery. 2001.
- [2] J. A. Gómez, "La asignatura de Sistemas Operativos I en el 2mil2", *Actas de las VII Jornadas de Enseñanza Universitaria de la Informática*, pgs. 533-540, Cáceres 2002.
- [3] A. Silberschatz y P. B. Galván, *Sistemas Operativos*, Addison-Wesley, 1999.
- [4] W. Stallings, *Sistemas Operativos*, Prentice may, 2001.
- [5] A. S. Tanenbaum, *Sistemas Operativos Modernos*, Prentice Hall, 2003.
- [6] R. Davoli y M. Goldweber, "New Directions in Operating Systems Courses Using Hardware Simulators", *Proceedings of SCS International Conference on Simulation and Multimedia in Engineering Education (ICSEE'03/WMC'03)*
- [7] F. Araque, F. Moral y J. A. Gómez, "Entorno Gráfico para el desarrollo de proyectos de sistemas operativos", *NOVATICA n° 144*, pgs. 52-56, Barcelona 2000.
- [8] Valgind en <http://valgind.org/>.
- [9] Linux Trace Toolkit, en <http://www.opersys.com/LTT>.