

Sistemas Operativos, un análisis a largo plazo de los procesos de innovación docente

Xavi Canaleta
Departament d'Enginyeria
La Salle, Universitat Ramon Llull
Barcelona
xavic@salleurl.edu

Resumen

A menudo los análisis que se realizan sobre los procesos de innovación docente suelen ser cortoplacistas. Muchas ponencias y artículos presentan resultados entre dos y tres cursos académicos como máximo. Sin discutir la validez de estos análisis, parece evidente que este espacio de tiempo no permite observar si los procesos de cambio se han consolidado y los resultados son estables o han evolucionado.

Por otro lado, un análisis del impacto de una innovación docente puede presentar resultados a corto plazo que no evidencian una mejora. Quizá sea necesario un espacio temporal mayor para la asimilación de los cambios en los procesos metodológicos y, sin esta visión, se corre el riesgo de no dar por válido una propuesta de mejora debido a un análisis precipitado y demasiado inmediato de los resultados obtenidos.

En esta ponencia se presenta el análisis histórico de una asignatura de tercer curso de la titulación de Ingeniería Informática en los últimos 13 cursos académicos. Con esta visión a largo plazo se pretende evaluar los diferentes procesos de cambio que se han introducido y el impacto que estos han tenido a lo largo del tiempo. A partir de aquí se podrán sacar conclusiones que deberían de ser aplicables a los procesos de innovación que se están planificando en la docencia universitaria de los estudios de ingeniería de la escuela.

Abstract

Often the analysis that are carried out on teaching innovation processes tend to be short-term. Many papers and articles present results between two and three academic years at the most. Without discussing the validity of these analysis, it seems evident that this period of time does not allow observing whether the processes of change have been consolidated and the results are stable or have evolved.

On the other hand, an analysis of the impact of a teaching innovation can present short-term results

that do not show an improvement. Perhaps a longer time space is needed to assimilate the changes in the methodological processes and, without this vision, there is a risk of not accepting a proposal for improvement as a result of a hasty and too immediate analysis of the results obtained.

In this paper we present the historical analysis of a subject of the third year of Computer Engineering degree in the last 13 academic years. With this long-term vision, we try to evaluate the different change processes that have been introduced and the impact they have had over time. From here, conclusions can be drawn that should be applicable to the innovation processes that are being planned in the university teaching of the engineering studies of the school.

Palabras clave

Innovación docente, evaluación de la docencia.

1. Motivación

La innovación es un proceso de cambio basado en la investigación que produce una mejora y es transferible. Cuando se realiza una innovación docente para poder evaluar si esta ha supuesto una mejora, se acostumbra a comparar resultados académicos, de satisfacción o mejora en la motivación de los estudiantes. Esta comparación se suele realizar con datos obtenidos antes de la aplicación del cambio y después (curso anterior y curso actual). Otro modo de realizar la validación, siempre que sea posible, es tener en paralelo un grupo donde se aplica el modelo tradicional y otro donde se realiza la innovación y se comparan los resultados obtenidos. Sea como fuere, el análisis de la mejora obtenida suele ser cortoplacista, presentando resultados de 2 o 3 cursos académicos a lo sumo.

Los cambios en las metodologías de aprendizaje introducidos en las asignaturas pueden provocar a corto plazo efectos más positivos o más negativos de lo que en su evolución realmente acaban siendo. Así

pues, este hecho podría llevar erróneamente a desestimar innovaciones que han obtenidos unos resultados por debajo de lo esperado en su primera implantación. Quizá necesitaban un cierto recorrido para acabar siendo adquiridas y asimiladas por los estudiantes. Del mismo modo, un cambio metodológico que da excelentes resultados en un primer momento podría derivar en cursos posteriores a resultados no deseados en su evolución.

Es por todo ello que este artículo tiene como objetivo hacer un análisis de la evolución que ha sufrido una asignatura en su trayectoria histórica (hasta donde se han podido recopilar datos) y que el proceso de reflexión de dicha evolución sea de utilidad (lecciones aprendidas) para el resto de asignaturas de los grados de ingeniería que tengan una tipología similar.

2. El marco de trabajo

En esta ponencia se analiza la asignatura de Sistemas Operativos desde el año 2004 al año 2017. Un factor importante en este estudio a tener en cuenta es que en nuestra universidad, en tercer curso del grado, existen 6 cursos académicos, concretamente el periodo que comprende desde el 2004-2005 al 2009-2010, donde la asignatura se ubicaba en un plan de estudios pre-Bolonia. Los 7 cursos posteriores, desde el 2010-2011 al 2016-2017, corresponden a la etapa Bolonia.

2.1. Etapa pre-Bolonia

Un sistema operativo es una capa de software que se ejecuta sobre un hardware para ofrecer una serie de servicios. Cada vez las funcionalidades de un sistema operativo incluyen más tareas y abarcan más servicios. El objetivo de la asignatura era visionar un sistema operativo en todas sus partes y, de esta manera, dotar al alumno de conocimientos sobre uno de los componentes más importantes en los sistemas informáticos en la actualidad.

SISTEMAS OPERATIVOS
<i>Semestre 1</i>
1. Introducción a los Sistemas Operativos
2. El núcleo de un sistema operativo
3. Planificación
4. Mecanismos de comunicación, sincronización y exclusión mutua.
<i>Semestre 2</i>
1. Sistema de Gestión de Memoria
2. Sistemas de Ficheros

Cuadro 1: Contenidos de Sistemas Operativos.

La asignatura de Sistemas Operativos era, en esta etapa, una asignatura anual con los contenidos que se describen en el Cuadro 1. Como se puede apreciar en los contenidos de la asignatura los conceptos corres-

pondientes al sistema de Entrada/Salida no se incluían en la asignatura dado que en el plan de estudios ya existía otra asignatura llamada Periféricos que ya contemplaba dichos conceptos.

La asignatura en este periodo se impartía en un formato presencial con 3 horas lectivas de clase a la semana. En esta etapa inicial pre-Bolonia la metodología docente era completamente tradicional. Las clases eran totalmente magistrales, con explicaciones teóricas y resolución de problemas. Adicionalmente, fuera de las horas lectivas, el alumno tenía que realizar una práctica de grandes dimensiones. Además existía una segunda práctica mucho más reducida sobre sistemas de ficheros.

Inicialmente, el sistema de evaluación se basaba en dos grandes calificaciones: por un lado las prácticas (con un peso del 30% de la nota final) y por otro el promedio de los exámenes finales semestrales (se ponderaba un 70%). Las dos partes debían aprobarse por separado para poder realizar la media ponderada y calcular la calificación final.

La práctica grande tenía como objetivo diseñar e implementar funcionalidades basadas en los conceptos del primer semestre de la asignatura. La segunda se focalizaba con los contenidos del tema 2 del segundo semestre. El resto de contenidos eran evaluados a través de los exámenes semestrales.

2.2. Etapa Bolonia

Durante el curso 2010-2011 se implantó en tercero de Grado de Ingeniería Informática el nuevo plan de estudios basado en Bolonia. Esto supuso un cambio en la estructura de la asignatura de sistemas operativos, puesto que las asignaturas de tercer y cuarto cursos pasaban a ser ahora semestrales. Así pues, tal y como muestra la Figura 1, la asignatura anual de Sistemas Operativos se descompuso en dos asignaturas semestrales llamadas Sistemas Operativos (de 5 créditos ECTS) y Sistemas Operativos Avanzados (de 4 créditos ECTS). Los contenidos de ambas asignaturas se correspondían bastante fielmente a los de los dos semestres de la asignatura pre-Bolonia.

Además de los cambios en cuanto a contenidos que se realizaron durante el período pre-Bolonia (y que serán detallados en el apartado de análisis de acciones y resultados) la separación de las dos asignaturas hizo que el sistema de evaluación de ambas fuera bastante diferente.

- Sistemas Operativos seguía evaluándose a partir de dos grandes ítems: examen semestral y la práctica. Se habían equilibrado los porcentajes en función del tiempo de dedicación. De este modo el examen final tenía un peso del 60% mientras que la práctica pasaba a ponderar un 40% de la calificación final.

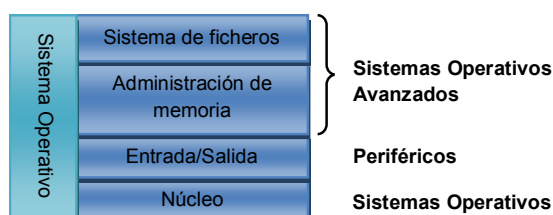


Figura 1: Relación entre contenidos y asignaturas.

- Sistemas Operativos Avanzados se evaluaba mediante la realización de dos exámenes para cada uno de los temas de la asignatura (70% de la nota final) y con la calificación obtenida en la práctica de sistemas de ficheros que se incorporó ya en los últimos cursos del sistema pre-Bolonia (30%).

El objetivo final del docente debe ser conseguir el máximo grado de aprendizaje de sus estudiantes (rendimiento académico) y, adicionalmente, tener un buen nivel de motivación y satisfacción de los mismos. En base a esta realidad y la situación inicial de la que se partió, a continuación se realiza un análisis de los resultados obtenidos para poder ver, a posteriori, si estos fueron consecuencia de las actuaciones de mejora que se fueron realizando durante este periodo de 13 cursos académicos.

3. Acciones: causa y consecuencia

La primera observación que cabe realizar es que la mayoría de acciones que se van a realizar van a tener como referencia la Figura 2. Esta figura expone de manera sintetizada los resultados académicos de los estudiantes de los 13 cursos. Los resultados reflejados son siempre resultados finales de la asignatura, incluyendo convocatoria y convocatoria extraordinaria de manera acumulada.

Como se puede observar inicialmente el problema más grave del que adolecía la asignatura es la gran cantidad de No Presentados (NP) que en los primeros años analizados de la etapa pre-Bolonia llegó a alcanzar tasas superiores al 50%. La causa principal y determinante de estos nefastos porcentajes de NP era la no presentación de la práctica. Durante el curso 2004-2005 el 96% de los alumnos calificados con NP era debido a no presentación de la práctica. De este 96% un 50% había aprobado el examen final y el resto, un 46% no se había presentado (no se puede afirmar, pero puede deducir que el motivo era que no encontraban sentido a presentarse al examen final si a ciencia cierta sabían que no llegarían a entregar la práctica con éxito).

Así pues, como se observará con los resultados posteriores las acciones iniciales fueron encaminadas a reducir estos porcentajes de abandono.

Dado que el punto crítico era la práctica de grandes dimensiones que en la fase pre-Bolonia estaba en la asignatura anual y que en la etapa Bolonia pasó a formar parte de la asignatura del primer semestre, los resultados que se comparan entre los dos periodos hacen referencia a las asignaturas de sistemas Operativos anual hasta 2009-2010 y Sistemas Operativos semestral a partir de 2010-2011.

Parece evidente que los periodos pre-Bolonia y Bolonia son difícilmente comparables dado que, aunque los temarios y objetivos de aprendizaje son los mismos en el primer semestre de una y otra etapa, existen otros factores que pueden condicionar los resultados obtenidos (la presencialidad, la aparición de otras asignaturas en el plan de estudios, etc.). De todos modos, la comparativa que se realiza no va tan orientada a mostrar los cambios entre etapas sino las innovaciones metodológicas que provocaron cierto cambio de ámbito. Es por ello que se mantiene esa visión global con las dos etapas conjuntamente.

3.1. Etapa pre-Bolonia

Las primeras actuaciones se centraron en mejorar el porcentaje de entregas de la práctica. Se detectaron dos problemas principales:

1. La mayoría de las herramientas necesarias para el desarrollo de la práctica no se explicaban en clase. Más allá de los mecanismos para la sincronización y la exclusión mutua (semáforos) y la creación de procesos en C (forks) pocos elementos más formaban parte de las clases lectivas, con lo cual el alumno debía documentarse y aprender el resto de las herramientas necesarias (signals, sockets, pipes, colas de mensajes...).
2. No se disponía de una documentación de soporte adecuada. Si bien es cierto que se facilitaba una buena bibliografía [4, 11, 12] no existía una documentación unificada y orientada al desarrollo de la práctica que facilitara al alumno el aprendizaje de todas las herramientas necesarias para su diseño e implementación.

De este modo, las primeras acciones se orientaron a mejorar estas dos deficiencias aplicándolas en el curso 2006-2007:

- A. Reestructuración de las clases lectivas incluyendo presentación de las herramientas con ejemplos
- B. Elaboración de diferentes manuales de soporte a las prácticas [5, 9, 10] focalizadas en las herramientas ad-hoc.
- C. Inclusión de sesiones de dudas a través de monitores de prácticas (becarios) que dan soporte durante la semana fuera de las clases lectivas.

Como puede apreciarse en la evolución de la Figura 2, las supuestas mejoras no fueron consistentes



Figura 2. Evolución de los resultados académicos de Sistemas Operativos

puesto que se pasó de un 45.6% de NP del cursos 2005-2006 a un 55.6% de abandono durante el 2006-2007 para reducirse esta tasa a un esperanzador 45.9%, el 2007-2008 y volver los dos siguientes cursos volvieron a sobrepasar el 50% de NP. ¿Eran las actuaciones realizadas erróneas? Seguramente no eran erróneas pero sí incompletas o insuficientes.

Una última reflexión sobre los datos académicos de la etapa pre-Bolonia: los “mejores” resultados en tasas de abandono en esta etapa son los obtenidos durante primer año 2004-2005, un 33.5% de NP. La primera constatación que se puede hacer es que en ese año la tasa de alumnos suspendidos era mucho más elevada que en cursos posteriores (concretamente del 7.1%), significativa comparándola con los de cursos posteriores, que son prácticamente inexistentes (todos los no aprobados son abandonos). Así pues sería mucho mejor hablar de un 40.6% de fracaso académico, agregando los porcentajes de no presentados y suspendidos. Sea como fuere, desde nuestro punto de vista tasas del 40% deberían ser inaceptables en un sistema universitario que se valore (y más en una asignatura de tercer curso).

3.2. Etapa Bolonia

Una vez finalizada la etapa pre-Bolonia sin unos resultados de mejora claros, se inició el cambio hacia el modelo del Espacio Europeo de Educación Superior. Un cambio significativo inicial fue que la asignatura de Sistemas Operativos del primer semestre se

dotó con 1 sesión de clase lectiva más debido al dato que en la escuela se decidió pasar de un modelo de sesión de duración de 1 hora a un modelo de sesión de 1 hora y 30 minutos. Así pues, aunque las sesiones en número eran las mismas (concretamente tres) ahora se pasaba de tener 3 horas de clase semanales a 4,5 horas (de hecho 4 horas en tiempo efectivo si se tienen en cuenta las pausas). Con este cambio y la incorporación del crédito ECTS como visión de carga total del alumno, se equilibró la misma y se distribuyeron los pesos de la evaluación de manera más coherente. Ahora la parte más conceptual (el examen final) y la práctica pasaban a tener el mismo valor: 50% de cada parte.

Así los cursos académicos 2010-2011 y 2011-2012 bajaron las tasas de no presentados pasando del 57.7% del 2009-2010 al 50.6% y 42.5% respectivamente. Se daban pasos pero se seguía lejos el objetivo planteado.

Curiosamente fue a través de la aportación de unos estudiantes observando los resultados del método PNI [3] que durante el curso 2012-2013 se introdujeron las sesiones de laboratorio. Analizando la satisfacción de los alumnos estos expresaban que uno de los puntos débiles para realizar las prácticas era que los alumnos no practicaban los diferentes recursos software hasta que los necesitaban. Así pues, por ejemplo, un alumno no programaba sockets hasta que los necesitaba para desarrollar un servidor dedicado que se le exigía diseñar e implementar en la práctica. Esto

suponía un doble esfuerzo: aprender el recurso y aplicarlo directamente sobre un entorno de desarrollo muy complejo.

Pero los estudiantes no actuaban así. Ellos intentaban aplicar directamente la herramienta sobre el problema complejo. Y si los estudiantes no procedían metódicamente, se debían poner los mecanismos para que lo hicieran. Era necesario conseguir que los estudiantes trabajaran los diferentes recursos antes de su aplicación a la práctica. Así que durante el curso 2012-2013 se crearon las sesiones de laboratorio. En una semana estándar:

- La primera sesión lectiva se dedicaba a presentar una herramienta en concreto (por ejemplo pipes) con explicación en clase y ejemplos de uso.
- Los estudiantes durante la semana podían preparar y revisar los ejemplos y hacer las consultas que creyeran oportunas.
- En la última sesión de la semana, al inicio de la clase se publicaba un ejercicio donde se debía realizar un programa en C que incluía el uso de la herramienta explicada. El alumno (individualmente o trabajando en pareja) tenía 1 hora y 20 minutos para diseñar e implementar el programa y colgarla en el campus virtual habilitado para ello.
- Inmediatamente al cierre de la tarea, se publicaba una posible solución al problema y en máximo una semana se daba la calificación a los alumnos del ejercicio.

La obligatoriedad de las sesiones y su impacto en la evaluación forzaban al alumno a trabajar las herramientas semanalmente. A partir de ese curso académico la nota final de la asignatura se calculaba:

$$\text{Nota} = 50\% \cdot \text{Conocimientos} + 50\% \cdot \text{Práctica}$$

$$\text{Conocimientos} = 65\% \cdot \text{Examen} + 35\% \cdot \text{Sesiones}$$

Las 12 sesiones estaban cuidadosamente planificadas para que se impartieran en el momento adecuado para el desarrollo de la práctica.

Los resultados del curso 2012-2013 hicieron bajar la tasa de NP al 39.7%. Pero hacía falta seguir refinando la dinámica para mejorar aún estos porcentajes. En el *feedback* que dieron los estudiantes del presente curso se detectaron dos puntos a considerar:

1. Las sesiones eran de gran utilidad como se comentaba en el 82% de las conclusiones del informe de prácticas.
2. Las sesiones no acababan de cumplir su objetivo, puesto que la exigencia de éstas hacía que los alumnos se desmotivaran en las primeras sesiones y ya no asistieran a las últimas.

A través de esta nueva iteración se introdujeron 2 cambios muy significativos en la evaluación de la asignatura:

- A. Si el objetivo final era que el estudiante practicara con las herramientas de sistemas antes de su uso en la práctica, era imprescindible motivar a que el alumno no abandonara las sesiones a la mitad del curso. Así pues se introdujo un cambio en la evaluación: si el alumno asistía como mínimo al 80% de las sesiones y estas le eran evaluadas se le garantizaba una nota mínima de 5, independientemente de la media obtenida. Las sesiones presentadas que no mostraran un esfuerzo significativo del estudiante serían consideradas como NP. De este modo, el alumno se motivaba seguir asistiendo y trabajando cada una de las herramientas durante todo el curso con independencia de los resultados obtenidos en las mismas.
- B. Analizando con detalle los objetivos de aprendizaje de la asignatura se consideró que la mayoría de ellos se cumplían con la realización de la práctica. Así se incluyó en el sistema de evaluación una frase que decía: “si el alumno, antes de la convocatoria ordinaria de examen final del primer semestre tiene la práctica entregada y aprobada según los criterios establecidos, se evaluará la posibilidad de quedar liberado del examen final en función de la calificación de la práctica, de su asistencia a las clases lectivas y de sus calificaciones en las sesiones de laboratorio”.

Analizando estas dos mejoras en el sistema de evaluación, claramente motivadoras para el estudiante, todos los elementos indicaban que la tasa de abandonos debería disminuir radicalmente durante el curso 2013-2014. La tasa ese año fue del 41.3% (frente al 39.7% del curso anterior). Ese curso académico, con 46 alumnos matriculados, tan solo 2 optaron a quedar liberados del examen final y lo consiguieron. Si se hubiera tirado la toalla en un análisis cortoplacista y revocado las medidas tomadas, no habría pasado lo que los próximos 3 años sucedió.

4. Resultados finales

Si se observan los resultados académicos de los tres siguientes cursos se puede observar el cambio sostenido que a partir de este segundo año de aplicación de la normativa se obtuvo. Los cursos 2014-2015, 2015-2016 y 2016-2017 las tasas de abandono descendieron del 40% anterior a valores del 24.15, 25.6% y 19.1% respectivamente. El número de alumnos que consiguieron quedar liberados de la realización del examen final fue del 25%, 41% y 42% durante esos tres años académicos. Pero, además, los alumnos que no consiguieron liberar el examen final

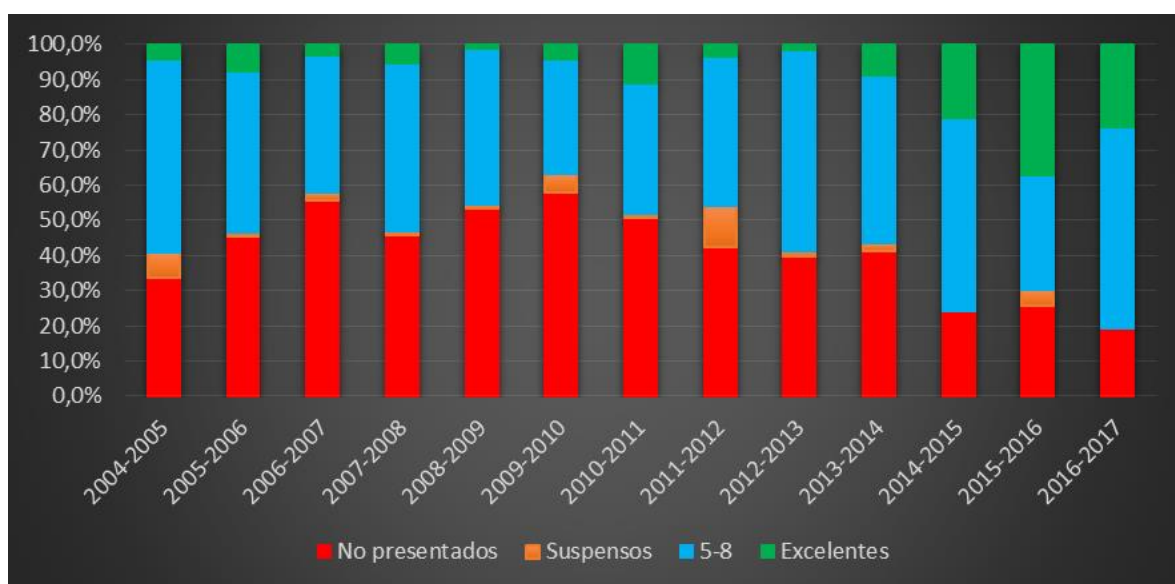


Figura 3. Evolución de calificaciones académicas de Sistemas Operativos

lo aprobaron y finalmente también entregaron la práctica con posterioridad, en convocatoria extraordinaria.

Si se analiza la información facilitada por la Figura 3, se puede observar que las mejoras que se han ido realizando durante toda la trayectoria de la asignatura han producido no tan solo una disminución significativa en cuanto a la tasa de abandonos sino también una mejora en las calificaciones académicas.

5. Otras consideraciones

En este apartado se quiere comentar dos aspectos importantes que pretenden dar consistencia y robustez a los resultados obtenidos. Por un lado está el método de evaluación de las prácticas y, por otro, la consideración de no incluir la asignatura de Sistemas Operativos Avanzados en el estudio realizado.

En primer lugar, siempre existe la tendencia a opinar que la eliminación del examen hace que el sistema de evaluación pierda objetividad y de este modo poner en duda la mejora de los resultados. En este punto cabe decir que en los últimos 5 cursos académicos el método utilizado para la evaluación de las prácticas es más estricto y sistemático que nunca, dada la importancia y repercusión de las mismas. Para la evaluación se utiliza una rúbrica que incluye tres ítems importantes:

1. Verificación de correcto funcionamiento de la práctica en ejecución: se realizan un conjunto de pruebas exhaustivas a todas las prácticas para validar su correcto funcionamiento.
2. Documentación: se evalúan tanto la memoria de prácticas realizada por los estudiantes como la calidad del código de las prácticas.

3. Entrevista individual: cada alumno es sometido a una entrevista personal donde debe dar respuesta a cualquier aspecto de diseño e implementación de la práctica. Las prácticas suelen tener un volumen entre 3.000 y 4.000 líneas de código y el alumno debe demostrar el conocimiento profundo del mismo.

De este modo existen datos objetivos que garantizan de forma sistemática e individual la evaluación del estudiante.

Por lo que refiere a la consideración de no incluir directamente la asignatura de Sistemas Operativos Avanzados en el estudio esta opción es por un doble motivo. Primero como se puede ver en la Figura 4, la asignatura de Sistemas Operativos Avanzados no presenta ninguna de las características problemáticas que se detectaron en la de Sistemas Operativos. En segundo lugar, sobre esta asignatura se han presentado diferentes ponencias a lo largo de sus diferentes cursos académicos que garantizan la buena salud docente que presenta [1, 2, 6, 7, 8].

Desde un punto de vista más global se pueden observar los cambios de la metodología docente de la asignatura de Sistemas Operativos si comparamos su estado inicial (curso 2004-2005) con la situación actual de la asignatura. La asignatura inicialmente estaba basada en la clase magistral del profesor, ya fuera para impartir las bases conceptuales o para la resolución de problemas. Además el estudiante debía diseñar e implementar una práctica de dimensiones considerables por su cuenta. Actualmente la asignatura exige un alto grado de presencialidad. Las sesiones de laboratorio son de asistencia obligatoria y con ellas se da una gran importancia a la evaluación continua. Además en el resto de sesiones se potencia la proactividad del estudiante puesto que la resolución de

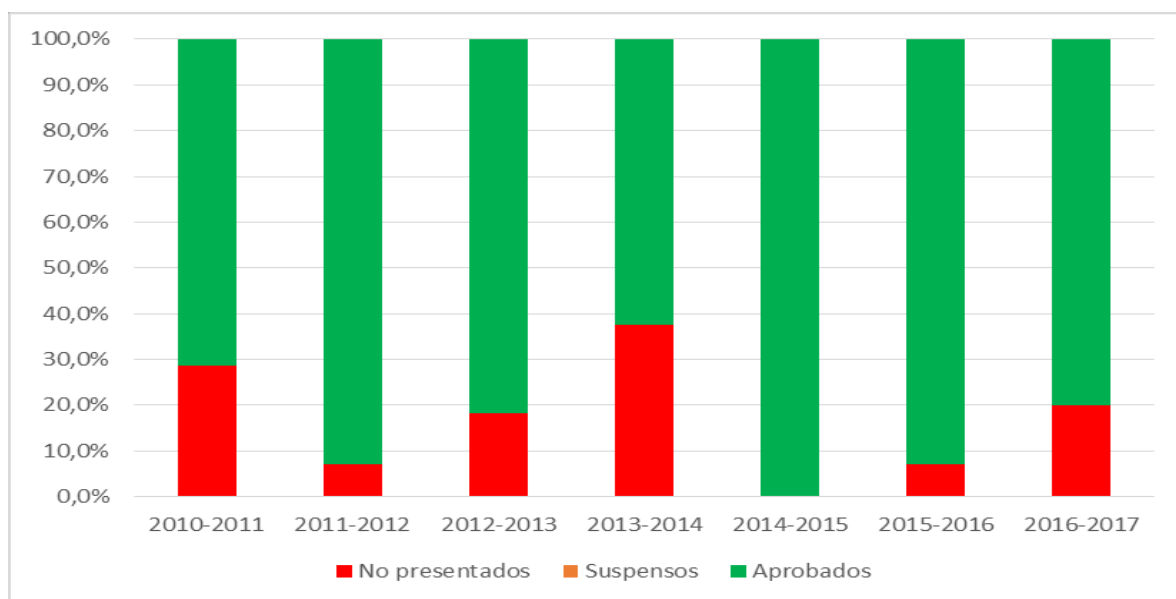


Figura 4. Evolución de calificaciones académicas de Sistemas Operativos Avanzados

problemas en clases se desarrolla a través de ellos y no del profesor, reduciendo de esta manera la intervención magistral del profesor a la explicación de los conceptos y herramientas imprescindibles para que los alumnos tengan las bases suficientes para realizar el resto de tareas. La mejora de los resultados académicos, tanto en la disminución de la tasa de abandonos como en el incremento de las calificaciones, es un hecho (Figura 3).

Finalmente, cabe destacar que este estudio se ha podido realizar seguramente gracias a la estabilidad del profesorado durante todo este periodo de tiempo. Además, el cambio de profesor titular podría ser una variable significativa que afectara a los resultados o mejoras obtenidas en este análisis. Como este hecho no se ha producido, este estudio gana en fiabilidad en este aspecto.

6. Conclusiones y líneas futuras

Con esta ponencia, y a la vista de los datos analizados en la trayectoria de la asignatura de Sistemas Operativos, se puede sacar un conjunto de lecciones aprendidas que se apuntaban como objetivos al inicio del artículo. Por un lado en cualquier proceso de innovación la precipitación a la hora de valorar resultados puede inducirnos a errores. Se necesita dar cierto margen de asimilación a los procesos de innovación docente. Y, a la vez, es necesario hacer un seguimiento constante de la evolución de los resultados porque mejoras significativas inmediatas no garantizan en modo alguno el éxito a medio o largo plazo.

Otro aspecto a destacar es que entre las cualidades que se deben exigir a un buen profesional en la docencia universitaria es la preocupación por la mejora continua de sus asignaturas. Esto implica no tan solo tener métricas válidas como puedan ser los resultados académicos, la satisfacción de los estudiantes o el *feedback* y sugerencias que estos puedan aportar, sino disponer del espacio temporal para su análisis y posible propuesta de acciones de mejora para los puntos débiles detectados.

Finalmente, volver a insistir en la importancia de una de las características de cualquier proceso de innovación: que este sea transferible. Si las aportaciones de esta ponencia permiten mejorar no tan solo una asignatura del Grado en Ingeniería Informática sino se pueden extender las buenas prácticas al resto de la titulación y, siendo más ambiciosos, pueden aportar algo de luz a otras escuelas universitarias el objetivo se puede dar por cumplido.

Los procesos de innovación deben estar siempre en la mente de cualquier profesional docente. Siguiendo en las mejoras de la asignatura durante el presente curso 2017-2018 se están incluyendo algunas sesiones de clase invertida junto con enseñanza entre pares con el objetivo de hacer más atractivas y dinámicas las pocas sesiones magistrales de conceptos que la asignatura actualmente incluye. No son grandes cambios pero se espera que estos signifiquen una mejora en la asignatura, esta vez en la satisfacción en el proceso de aprendizaje de los estudiantes.

Referencias

- [1] Xavi Canaleta y Joan Navarro. Herramientas de soporte al aprendizaje de sistemas de ficheros, En *Actas de las XVI Jornadas de Enseñanza Universitaria de Informática, Jenui 2010*, pp. 413-419, Santiago de Compostela, julio 2010.
- [2] Xavi Canaleta, David Vernet y Joan Navarro. Metodología on demand para el desarrollo de la asignatura de Sistemas Operativos Avanzados. En *Actas de las XIX Jornadas de Enseñanza Universitaria de Informática, Jenui 2013*, pp. 167-173, Castelló, julio de 2013.
- [3] Xavi Canaleta, Joan Navarro, Xavi Solé, David Vernet y Pau López. Método no formal para la evaluación de la docencia aplicada al Grado de Ingeniería Informática, En *Actas de las XX Jornadas de Enseñanza Universitaria de Informática, Jenui 2014*, pp. 445-449, Oviedo, julio 2014.
- [4] S.P. Harbison y G.L. Steele. C - A Reference Manual, Prentice Hall, 5a edición, 2002.
- [5] Hugo Meza. Manual de prácticas para Sistemas Operativos, Publicacions La Salle, septiembre 2005.
- [6] Joan Navarro, Xavi Canaleta y Andreu Sancho. Sistemas Operativos Avanzados: de la clase magistral al entorno colaborativo. En *Actas de las XVII Jornadas de Enseñanza Universitaria de Informática, Jenui 2011*, pp. 143-150, Sevilla, julio 2011.
- [7] Joan Navarro, Xavi Canaleta y Xavi Salada. LSscheduling: herramienta interdisciplinar de soporte docente para la asignatura de Sistemas Operativos. En *Actas de las XVIII Jornadas de Enseñanza Universitaria de Informática, Jenui 2012*, pp. 365-372, Ciudad Real, julio 2012.
- [8] Joan Navarro, Xavi Canaleta, David Vernet, Xavi Solé, Virginia Jiménez-Ruano y Nuria Costa. Motivación, desmotivación, sobremotivación, y daños colaterales. En *Actas de las XX Jornadas de Enseñanza Universitaria de Informática, Jenui 2014*, pp. 467-474, Oviedo, julio 2014.
- [9] Jordi Salvador. Programació en C per a sistemes UNIX, Publicacions La Salle, Segunda edición, Septiembre 2014.
- [10] Jordi Salvador. Introducció al llenguatge de programació C, Publicacions La Salle, Julio 2008.
- [11] R. Stevens y A.A. Rago. Advanced Programming in the UNIX Environment, Addison-Wesley Professional, 2a edición, 2008.
- [12] R. Stevens, B. Fenner y A.M. Rudoff. UNIX Network Programming, volum 1: "The sockets Networking API", Addison-Wesley Professional, 3a edición, 2004.