

Una experiencia de clase invertida en la enseñanza de la programación

Ángeles López

Depto. de Ingeniería y Ciencia de los Computadores
Universitat Jaume I
Castellón
angeles.lopez@uji.es

Mar Marcos

Depto. de Ingeniería y Ciencia de los Computadores
Universitat Jaume I
Castellón
mar.marcos@uji.es

Resumen

El contexto de este trabajo es la asignatura Programación I del Grado en Diseño y Desarrollo de Videojuegos de la Universitat Jaume I. El plan de estudios de este grado combina una formación en conocimientos de las TIC con contenidos de arte, diseño conceptual, comunicación, narrativa y producción, buscando un perfil que integre las competencias necesarias para el diseño y desarrollo de videojuegos. La asignatura Programación I se incluye en el primer semestre del primer curso. A la vista de las bajas tasas de rendimiento y éxito de la asignatura, en comparación con las demás asignaturas de primero, se puede concluir que es la que más dificultades plantea al estudiantado medio que ingresa en el grado. La asignatura Programación I se ha venido impartiendo desde sus inicios con una metodología clásica, en la que las profesoras imparten los contenidos teórico-prácticos del temario, proponiendo al estudiantado la resolución de ejercicios adicionales en su tiempo de trabajo no presencial. Recientemente hemos detectado un ligero empeoramiento de las tasas de rendimiento y éxito. Una posible hipótesis que explicaría este hecho es que el estudiantado no está dedicando el tiempo requerido para completar el trabajo no presencial planificado. Para corroborar esta hipótesis e intentar corregir la situación actual, en el curso 2017/18 hemos realizado cambios en la metodología docente de la asignatura, aplicando la filosofía de la clase invertida. En este trabajo presentamos el detalle de la metodología utilizada, así como las observaciones que hemos recogido y nuestras reflexiones al respecto.

Abstract

The context of this work is the Programming I course of the Degree in Design and Development of Videogames of Universitat Jaume I. The study plan of this degree combines a training in ICT knowledge with contents of art, conceptual design, communication, na-

rative and production, seeking a profile that integrates the necessary skills for the design and development of videogames. The Programming I course is included in the first semester of the first course. In view of the low performance and success rates of the course, compared with the rest of first-year courses, it can be concluded that it is the course that poses the greatest difficulties for the average student entering the degree. The Programming I course has been taught since its inception with a classical methodology, in which teachers impart the theoretical and practical content of the syllabus, proposing to the student the resolution of additional exercises in their time outside the classroom. Recently we have detected a slight worsening of the performance and success rates. A possible hypothesis that would explain this fact is that the students are not dedicating the required time to complete the scheduled out-of-class work. To corroborate this hypothesis and try to correct the current situation, in this academic year 2017/18 we have made changes in the teaching methodology of the course, applying the philosophy of the flipped classroom. In this paper we present the details of the methodology used, as well as the observations we have gathered and our reflections on these.

Palabras clave

Programación, Clase invertida, Proceso de enseñanza-aprendizaje.

1. Introducción y motivación

El contexto de este trabajo es la asignatura Programación I del Grado en Diseño y Desarrollo de Videojuegos (GDDV) de la Universitat Jaume I (UJI). Se trata de un grado de reciente creación y fue uno de los dos primeros grados oficiales dedicados a la temática de los videojuegos que se implantaron a nivel estatal. El plan de estudios del GDDV de la UJI combina una forma-

ción en conocimientos de las TIC (que cubre las competencias básica propias del Grado en Ingeniería Informática) con contenidos de arte, diseño conceptual, comunicación, narrativa y producción, con el fin de que los egresados tengan un perfil que integre los conocimientos teóricos y prácticos necesarios para el diseño y desarrollo de videojuegos. La asignatura Programación I junto con Informática Básica y Programación II se integra en el primer curso del GDDV, siendo estas tres asignaturas las únicas de informática del primer curso. A la vista de las bajas tasas de rendimiento y éxito de Programación I, en comparación con el resto de asignaturas de primer curso, se puede concluir que es la asignatura que más dificultades plantea al estudiantado medio que ingresa en el GDDV. No es de extrañar porque, como es bien sabido, aprender a programar es una tarea muy difícil, que implica la adquisición no solo de nuevos conocimientos sino también de estrategias y habilidades prácticas complejas [6].

La asignatura Programación I del GDDV se ha venido impartiendo desde sus inicios con una metodología clásica, en la que las profesoras imparten los contenidos teórico-prácticos del temario mediante clase magistral, proponiendo al estudiantado la resolución de ejercicios adicionales en su tiempo de trabajo no presencial. Recientemente hemos detectado un ligero empeoramiento de las tasas de rendimiento y éxito de la asignatura, ya de por sí más bajas que las del resto de asignaturas de primero. Una posible hipótesis que explicaría este hecho es que el estudiantado no está dedicando el tiempo necesario para la realización del trabajo no presencial planificado. Para corroborar esta hipótesis e intentar corregir la situación actual, en este curso 2017/18 hemos realizado cambios en la metodología docente de la asignatura aplicando la filosofía de la clase invertida.

Gracias a la investigación educativa se sabe que los estudiantes aprenden más por lo que ellos hacen que por lo que hace el profesor, de modo que una función fundamental del profesor sería facilitar las condiciones necesarias para que aprendan por sí mismos [2, 7]. La clase invertida [1, 2, 5], o *flipped classroom*, propone reducir el tiempo dedicado a la clase magistral, trasladando fuera del aula la exposición de contenidos, esto es, el trabajo a niveles cognitivos más bajos (recordar, comprender), para dedicar más tiempo en el aula a realizar tareas donde se trabajan niveles cognitivos más altos (aplicar, analizar, evaluar, crear), que son las que suelen presentar más dificultad para los estudiantes.

Hay una gran cantidad de publicaciones relacionadas con esta metodología y que proponen una variedad de estrategias para mejorarla. Una de estas estrategias es la enseñanza justo a tiempo [4, 8], o *JiTT (just-in-time teaching)*, donde el profesor diseña las sesiones de clase a partir de las soluciones de los estudiantes a una

serie de tareas que se realizan previamente. Esto permite diseñar actividades mucho más dirigidas (corregir los errores más habituales, trabajar soluciones alternativas, etc.).

En el diseño de la nueva metodología docente para la Programación I del GDDV hemos adoptado una versión de la clase invertida con actividades semanales que permiten aplicar la estrategia JiTT. En este trabajo presentamos el detalle de la metodología utilizada, así como las observaciones que hemos recogido y nuestras reflexiones, durante y al finalizar la actividad docente. Para terminar, presentamos nuestras conclusiones a la vista de estas observaciones y de los resultados de los estudiantes.

2. Descripción de la asignatura

La Programación I del GDDV de la UJI es una asignatura de formación básica que se imparte en el primer semestre del primer curso del grado y es, en la mayoría de los casos, el primer contacto de los estudiantes con las metodologías y tecnologías de la programación. Como resultados del aprendizaje de la asignatura se espera que el estudiante conozca: los tipos de datos básicos de un lenguaje de programación, las estructuras básicas de control (iteración y selección), las estructuras de datos básicas (cadenas, vectores, registros y matrices), la definición y uso de funciones, una introducción a clases y objetos, y la entrada y salida basada en ficheros de texto. Asimismo, se espera que el estudiante sea capaz de implementar programas que hagan uso de estructuras de control de iteración y/o selección, posiblemente anidadas, para manejar las estructuras de datos estudiadas. En cuanto a la estructuración de la asignatura, tiene 6 créditos y su carga de trabajo se distribuye en 31 horas de clases de teoría, 23 horas de prácticas en aula informática, 6 horas de evaluación, y un total de 90 horas de trabajo no presencial (trabajo personal y preparación de exámenes). En cuanto al sistema de evaluación, está dividido a partes iguales entre un examen y la evaluación continua.

Destacar que el diseño de algunas de las asignaturas del ámbito de la informática del GDDV de la UJI se basó en gran medida en la dilatada experiencia del profesorado de la UJI en la docencia del Grado en Ingeniería Informática (GII). En el caso de Programación I, el diseño de la asignatura es idéntico al de la asignatura homónima del GII, excepto por el sistema de evaluación, que en el caso del GII está dividido entre un examen, la evaluación de prácticas y la evaluación continua. A partir de la experiencia en la docencia del GII, la dirección del GDDV eligió utilizar el lenguaje de programación Python en la docencia de Programación I. En consonancia con esto, como bibliografía básica se decidió utilizar el libro “Introducción a la programación

con Python 3” de Marzal *et al* [3].

En cuanto al perfil del alumnado del GDDV, la mayor parte de los estudiantes proceden del bachillerato, aunque también hay estudiantes procedentes de ciclos formativos de grado superior, titulados universitarios y estudiantes que han accedido mediante la prueba de acceso para mayores de 25 años (el orden indica un número decreciente de estudiantes). Cabe mencionar también que actualmente la mayoría de estudiantes proceden del bachillerato de Ciencias o de ciclos formativos y titulaciones científico-tecnológicas.

De acuerdo con los criterios de planificación de la UJI, la asignatura cuenta desde hace varios cursos con 1 grupo de teoría y 3 grupos de laboratorio/prácticas. Además, desde el curso 2015/16 hasta la fecha hemos contado con 1 grupo de teoría y 1 grupo de prácticas adicionales como acciones de mejora docente, lo que nos ha dado margen para dedicar estos grupos adicionales a estudiantes de 2ª matrícula (y posteriores). Esto nos ha permitido plantear las clases de teoría para los estudiantes de 2ª matrícula de manera diferente, prestando más atención a la resolución de ejercicios de programación en el aula y a la corrección de los errores típicos de los programadores inexpertos. Los resultados positivos de esta experiencia han sido el germen del cambio en la metodología docente de la asignatura.

3. Diseño de la metodología docente

Como ya hemos mencionado, en el diseño de la nueva metodología docente de la asignatura hemos adoptado la filosofía de la clase invertida y de la estrategia JiTT. Nuestra principal motivación ha sido pautar de la manera más clara posible el trabajo que el estudiante debe realizar cada semana, con el fin de lograr el máximo aprovechamiento de las clases de teoría y de progresar adecuadamente en la asignatura.

En nuestra versión de la clase invertida proponemos unas actividades previas a las **clases de teoría**, consistentes en la lectura de material del libro y en la resolución de un pequeño número (2 o 3) de ejercicios básicos. El estudiante debe entregar sus soluciones en un plazo que acaba el día antes del de la clase de teoría, lo que nos permite revisarlas y diseñar la clase en función de las mismas (estrategia JiTT). Además, las soluciones a los ejercicios se publican el día antes de la clase, una vez concluido el plazo de entrega, con el fin de que estudiante pueda cotejarlas con las suyas y hacer una primera autoevaluación.

Dando por supuesto que los estudiantes han trabajado en las actividades previas, las sesiones de teoría se dedican en primer lugar a la explicación de las soluciones publicadas y a la evaluación de las alternativas de

los estudiantes. Destacar que se dan explicaciones teóricas solo cuando alguno de los estudiantes lo solicita, o cuando la complejidad de algún aspecto concreto lo requiere. El resto de la clase se dedica a la resolución de ejercicios de programación de mayor complejidad (y de complejidad creciente). De este modo, el número de ejercicios que se resuelven en el aula es mucho mayor, comparado con una metodología basada en la clase magistral.

La metodología de las **clases de prácticas** prácticamente no ha cambiado. En este caso también se pautan los boletines de prácticas en los que el estudiante debe trabajar cada 2 (o 3) semanas, se estipula un plazo de entrega y se publican las soluciones una vez concluido dicho plazo. En las clases de prácticas se deja mayor libertad al estudiante, permitiendo que avance más o menos rápidamente en función de su nivel de destreza.

Las actividades previas, los boletines de prácticas, las actividades de clase y las soluciones se han ido publicando en un curso del Aula Virtual de la UJI¹, basada en la herramienta *Moodle*. Para facilitar la planificación del trabajo del estudiante y dejar constancia del ritmo al que se avanza en la asignatura, se ha utilizado un formato por semanas. Lógicamente, una parte importante del curso Moodle son las tareas para las entregas de los estudiantes así como los enlaces a las soluciones de los distintos ejercicios. Otro aspecto a destacar es que hemos utilizado la funcionalidad de Moodle para el rastreo de *finalización de actividades*, que permite al profesor configurar criterios de finalización dentro de una actividad específica, como puede ser la entrega de una tarea o la visualización de un documento.

En cuanto al **sistema de evaluación**, se ha mantenido la separación entre examen y evaluación continua, dividiendo esta última en una parte de teoría y otra de prácticas. Concretamente, un 50 % de la calificación corresponde al examen, un 20 % a la evaluación continua, parte teoría, y un 30 % la evaluación continua, parte prácticas. La evaluación continua, parte teoría, está basada en una prueba de menor complejidad que el examen que se realiza en el último mes del semestre. La evaluación continua, parte prácticas, se basa en dos pruebas que se realizan al terminar la 2ª y 4ª prácticas, respectivamente. Es importante destacar que la calificación se basa exclusivamente en pruebas de evaluación y que la entrega de las actividades previas y de las prácticas no se puntúa. En nuestra experiencia, asociar una puntuación a entregas de código a menudo resulta en un incremento del número de entregas por parte de estudiantes que no han realizado el trabajo.

¹<http://aulavirtual.uji.es>

4. Resultados y análisis

Dados los cambios planteados en la metodología docente, durante el semestre hemos llevado a cabo un seguimiento de la realización de las actividades por parte de los estudiantes. Esto se ha hecho usando el rastreo de finalización de actividades de Moodle mencionado anteriormente. Concretamente, hemos recopilado información sobre los siguientes aspectos para cada estudiante:

- Si ha hecho entrega de su trabajo en cada una de las actividades previas y de las prácticas.
- Si ha consultado las soluciones de cada una de las actividades previas y de las prácticas.

La finalización de actividades se ha ido comprobando semanalmente de manera global con el fin de animar a los estudiantes a continuar trabajando en la asignatura de manera regular. Al terminar la docencia de la asignatura se ha hecho un análisis más detallado cuyos resultados se presentan en las subsecciones siguientes. Otro aspecto a tener en cuenta relacionado con el rastreo de finalización es que, en el caso de las soluciones, el rastreo sólo refleja si el estudiante ha visualizado/abierto el documento.

Adicionalmente, al final del semestre hemos realizado una encuesta anónima para recopilar la valoración de los estudiantes acerca de diversos aspectos de la asignatura y de la metodología utilizada. La encuesta la respondieron un total de 19 estudiantes, de los cuales 9 eran de primera matrícula, y 10 de segunda o posterior.

4.1. Rastreo de finalización de actividades

Actividades previas En las figuras 1 y 2 se muestran los porcentajes de entrega de las actividades previas y de consulta de las soluciones, respectivamente. En ambos casos, el porcentaje claramente va decreciendo a medida que avanza el curso. En ambas gráficas podemos observar también la diferencia entre el grupo de estudiantes de primera matrícula (T1) y el grupo de estudiantes repetidores (T2). Hay un mayor porcentaje de estudiantes repetidores que entregaron y consultaron las soluciones de cada actividad. Creemos que esta diferencia se debe a que hay un porcentaje mayor de estudiantes repetidores que son conscientes de la necesidad de realizar este trabajo no presencial, y de que es responsabilidad suya. Aclarar que aunque algunos de los estudiantes del grupo de repetidores habían cursado la asignatura alguna vez con una metodología centrada en la resolución de problemas, la mayoría de ellos lo habían hecho con la metodología de clase magistral utilizada anteriormente.

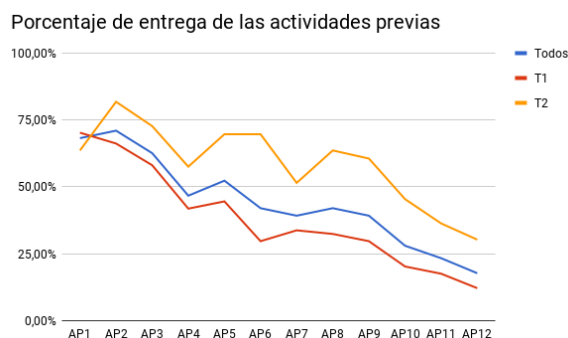


Figura 1: Porcentaje de entrega de cada una de las 12 actividades previas: en rojo, el grupo de estudiantes de primera matrícula, en amarillo, el grupo de repetidores y en azul, todos.

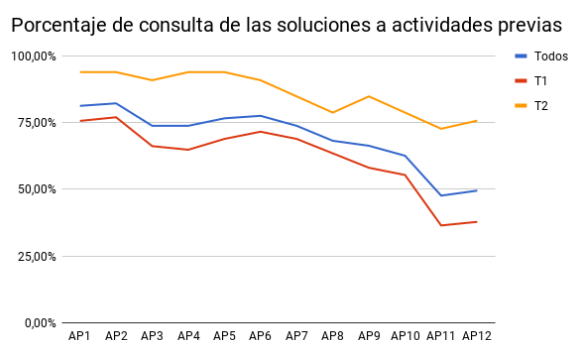


Figura 2: Porcentaje de consulta de las soluciones de cada una de las 12 actividades previas.

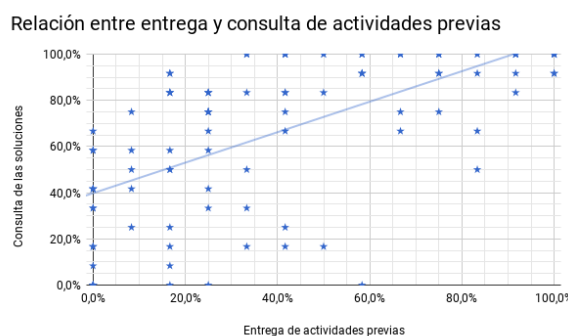


Figura 3: Relación entre el porcentaje de entrega y el porcentaje de consulta de las soluciones de cada uno de los estudiantes.

En la figura 3 se puede ver la relación entre porcentaje de entregas y de consulta para cada estudiante. Se puede observar que hay estudiantes que entregan más que consultan, y viceversa, pero no se puede ver cuántos estudiantes hay en cada caso porque los puntos se superponen. Aunque se puede ver en la gráfica la línea de tendencia, nos interesa saber el promedio de entregas de los estudiantes con un determinado porcentaje de consultas. En la figura 4 se pueden observar los resultados. Los estudiantes que consultaron el 50 % de las soluciones, entregaron una media de aproximadamente el 30 % de las actividades previas, mientras que los estudiantes que consultaron el 100 %, entregaron alrededor del 90 % de media. También se muestra el número de estudiantes de cada caso. Un dato interesante es que el caso más numeroso, con diferencia, es el de los estudiantes que consultaron el 100 % de las soluciones.

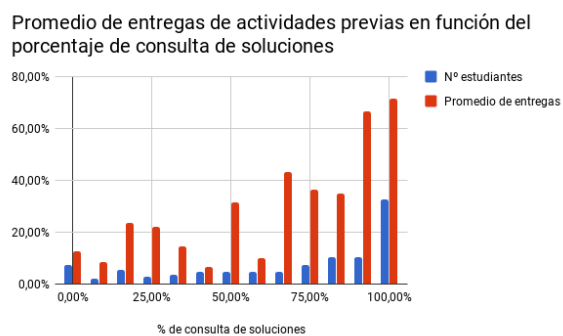


Figura 4: Promedio de los porcentajes de entrega en función del porcentaje de consulta de las soluciones de las actividades previas.

También hemos elaborado los cálculos a la inversa. En la figura 5 se muestra el promedio de consultas de las soluciones por parte de los estudiantes que entregaron un determinado porcentaje de actividades previas. Se puede observar que, incluso los estudiantes que sólo entregaron una de las 12 actividades consultaron al menos el 50 % de las soluciones.

Prácticas Hemos realizado un estudio similar para la entrega y consulta de las soluciones de la prácticas. En las figuras 6 y 7 se muestran los porcentajes de entrega y de consulta de las soluciones, respectivamente. En este caso, no hubo entrega de la primera práctica, de modo que esta información no aparece en las gráficas. Sin embargo sí que recopilamos la consulta de las soluciones desde la primera práctica que, en términos generales fue mayor si cabe que la consulta de las soluciones de las actividades previas. No es de extrañar si tenemos en cuenta que las soluciones de las actividades previas se mostraban en la clase de teoría correspondiente y, además, constan de entre 2 y 4 ejer-

Promedio de consulta de soluciones en función del porcentaje de entregas de actividades previas

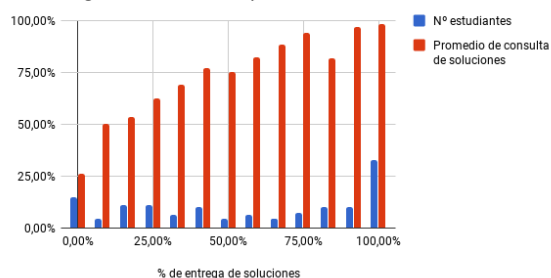


Figura 5: Promedio de los porcentajes de consulta de las soluciones en función del porcentaje de entregas de las actividades previas.

cicios cortos a diferencia de las prácticas que constan de una media de 10 ejercicios y de mayor dificultad. La diferencia entre el grupo de estudiantes repetidores y el de primera matrícula es menor que en el caso de las actividades previas.

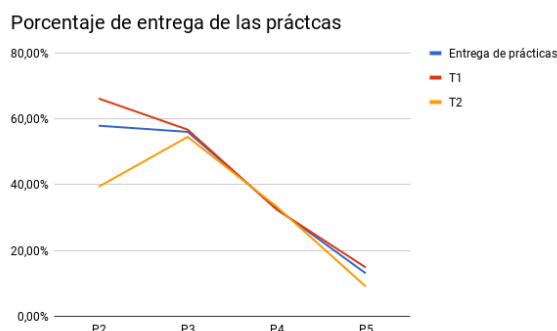


Figura 6: Porcentaje de entregas de cada una de las 5 prácticas de laboratorio.

En la figura 8 se resume el aprovechamiento de cada una de las 5 prácticas. El porcentaje de consultas es muy superior al de entrega, y hemos calculado un porcentaje de aprovechamiento como la media de ambos (salvo en el caso especial de la práctica 1).

También nos interesa saber en qué medida los estudiantes que realizan las prácticas también realizan las actividades previas y viceversa. En la gráfica de la figura 9 se muestra el aprovechamiento (calculado como el promedio de entregas y consultas) de las actividades previas y de las prácticas de cada estudiante. La línea de tendencia muestra una clara correlación entre ambos porcentajes de aprovechamiento. Una observación interesante es que los repetidores en general parecen haber dado más importancia a las actividades previas que el resto de los estudiantes (la línea de tendencia está por encima de la identidad).

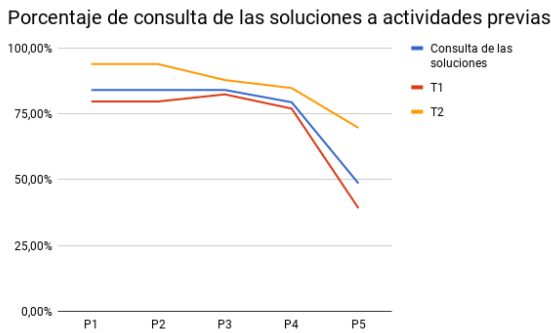


Figura 7: Porcentaje de consulta de las soluciones de cada una de las 5 prácticas de laboratorio.

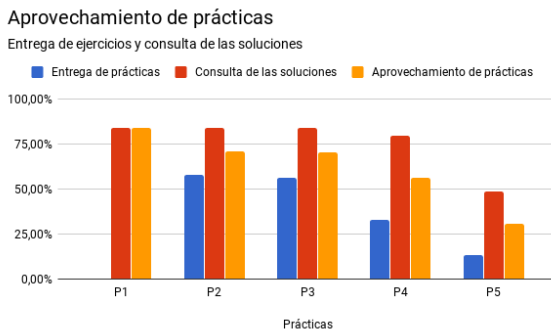


Figura 8: Comparación de los porcentajes de entrega y de consulta de las soluciones de cada una de las 5 prácticas de laboratorio (en la primera práctica no hubo entrega) y promedio de ambos.

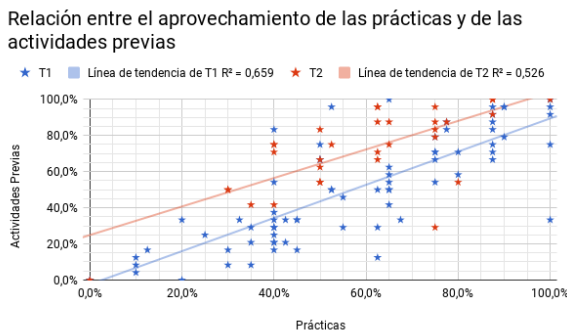


Figura 9: Relación entre el aprovechamiento de las actividades previas y el aprovechamiento de las prácticas de cada estudiante (promedio de las entregas y la consulta de las soluciones).

4.2. Encuesta anónima

La encuesta nos permitió también conocer la cantidad de horas que dedican nuestros alumnos a realizar estas actividades. Concretamente, les preguntamos cuántas horas semanales en promedio dedicaban a cada una de estas tareas: realizar las actividades previas, repasar la teoría, resolver las prácticas y resolver ejercicios. En la figura 10 se puede ver el promedio de las horas dedicadas por los estudiantes que respondieron a la encuesta, en porcentaje. El promedio del total de horas indicado por los estudiantes es de 7 horas semanales. Para hacer estos cálculos hemos tenido que eliminar dos de las respuestas que sumaban más de 40 horas, por entender que se debe a una mala interpretación de la pregunta, y que por tanto consideramos *outliers*. Lo más interesante de la figura 10 es comprobar que la resolución de las prácticas sigue siendo la tarea a la que los estudiantes dedican más tiempo.



Figura 10: Distribución de las horas semanales dedicadas a la asignatura, calculada a partir de los resultados de la encuesta.

Aprovechamos la encuesta para preguntar a los estudiantes sobre la dificultad de los contenidos de teoría y de las prácticas, así como de las pruebas de evaluación correspondientes. En la figura 11 se resumen sus respuestas. Como se puede ver, la apreciación de los estudiantes sobre la dificultad de los contenidos y de las pruebas de evaluación sigue una distribución normal centrada en el valor medio.

4.3. Datos de presentados y tasas de éxito y rendimiento

Durante los cursos anteriores habíamos detectado una disminución en el porcentaje de estudiantes presentados al examen final de la primera convocatoria. Aún con la nueva metodología, el descenso se ha mantenido en el grupo de los estudiantes de primera matrícula (el porcentaje de presentados ha pasado del 77 % al 60 %). Sin embargo, parece que la nueva metodología ha animado a los repetidores a seguir la asignatura hasta presentarse al examen final (ha pasado del 71 % al 85 %).

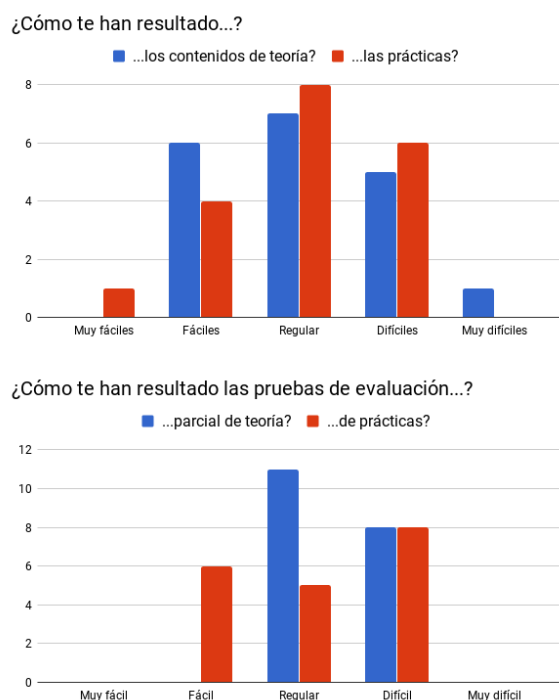


Figura 11: Opinión de los estudiantes acerca de la dificultad de los contenidos de teoría y las prácticas (arriba) y las pruebas de evaluación continua de teoría y de prácticas (abajo).

En cuanto a la tasa de éxito, se ha mantenido estable en términos globales. Sin embargo, se aprecia una ligera mejora en el grupo de repetidores y un empeoramiento en el grupo de estudiantes de primera matrícula. Una primera valoración es que la nueva metodología, que demanda un trabajo individual regular para poder seguir adecuadamente la asignatura, ha tenido que ver con el hecho de que estos estudiantes se sientan poco preparados para afrontar el examen.

4.4. Rastreo de finalización de actividades vs. calificaciones de evaluación continua

En la figura 12 se muestra la relación entre el porcentaje de entrega de actividades previas y la calificación de evaluación continua de cada estudiante. Se muestra en rojo los resultados del grupo de repetidores y en azul los del grupo de primera matrícula. De nuevo se aprecia que los repetidores por término medio han realizado más actividades y han conseguido mejores calificaciones. Las líneas de tendencia, además, nos permiten ver que en términos generales la entrega de actividades previas y el rendimiento ha sido mayor en el grupo de repetidores, mientras que el grupo de primera matrícula hay más diversidad de casos. Cabe resaltar el caso de los estudiantes que, a pesar de haber

realizado el 100 %, o casi, de las entregas, han obtenido malas calificaciones. Esto se debe a que el hecho de realizar las entregas no significa que éstas sean correctas. También hay estudiantes que consiguen aprobar a pesar de tener un porcentaje de entregas bajo. Esto es debido a que las entregas no eran obligatorias ni aportan puntos a la calificación, por lo que ésta no se ve influenciada en absoluto por este aspecto y depende exclusivamente del rendimiento de los estudiantes en las pruebas de evaluación continua.

Relación entre entrega de actividades previas y notas de evaluación continua

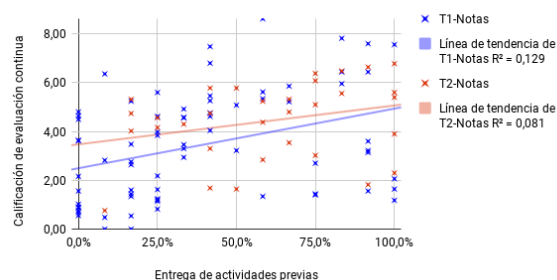


Figura 12: Relación entre el porcentaje de entrega de las actividades previas y las calificaciones de la evaluación continua para cada estudiante.

Si bien un único aspecto no es determinante en relación con las calificaciones, si tenemos en cuenta más aspectos, la relación se estrecha. Así, en la figura 13 se puede ver la relación entre el porcentaje de aprovechamiento (calculado como el promedio de la entrega de las actividades previas y de las prácticas y la consulta de las soluciones de ambas) y las calificaciones de la evaluación continua.

Relación entre porcentaje de aprovechamiento y notas de evaluación continua

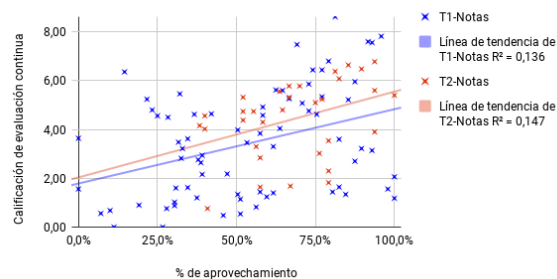


Figura 13: Relación entre el porcentaje de aprovechamiento de las tareas (promedio de las actividades previas y de las prácticas) y las calificaciones de la evaluación continua para cada estudiante

5. Conclusiones

En este trabajo presentamos un nuevo diseño de la metodología docente para la asignatura Programación I del GDDV de la UJI, basado en la clase invertida y con una estrategia JiTT. Con el fin de evaluar la respuesta de los estudiantes ante la aplicación de la nueva metodología hemos llevado a cabo una serie de acciones de seguimiento utilizando la herramienta *Moodle* y hemos analizado las relaciones existentes entre los datos recopilados.

Un primer análisis de los resultados revela hallazgos interesantes como es el interés de los estudiantes en las soluciones publicadas independientemente de si han realizado o no la entrega correspondiente. De hecho, incluso los estudiantes que realizaron menos de un 10% de las entregas consultaron un 50% o más de las soluciones. Otra observación interesante es que, de acuerdo con la información proporcionada por los estudiantes que respondieron a la encuesta, la actividad a la que más tiempo dedican es la resolución de las prácticas. Cabría preguntarse por qué no dedican una cantidad de tiempo comparable a la realización de otros ejercicios.

Con respecto a las diferencias entre estudiantes de primera matrícula y repetidores, cabe destacar que los segundos parecen haber dado más importancia a las actividades previas que los primeros. Con otras palabras, parece que la metodología de clase invertida se adapta mejor a sus necesidades. Sin embargo, basándonos en algunas de las respuestas de la encuesta, parece que la percepción de los estudiantes de primera matrícula es que con esta metodología las clases de teoría dejan de tener sentido.

Como mejora de cara al próximo curso, vamos a priorizar el transmitir mejor la filosofía de la clase invertida, haciendo hincapié en que la sustitución de las clases magistrales/expositivas por otro tipo de actividades no resta validez ni importancia a las clases de teoría. En lo que al sistema de evaluación se refiere, vamos a volver a evaluar la conveniencia de puntuar la entrega de las actividades previas y las prácticas.

Referencias

- [1] Jonathan Bergmann y Aaron Sams. *Flip Your Classroom. Reach Every Student in Every Class Every Day*. ASCD & ISTE, 2012. ISBN: 978-1564843159.
- [2] Mercedes Marqués. “Qué hay detrás de la clase al revés (flipped classroom)”. En: *XXII Jornadas de Enseñanza Universitaria de la Informática. JENUI 2016*. (Almería). Jul. de 2016, págs. 77-84. ISBN: 978-84-16642-30-4.
- [3] Andrés Marzal Varó, Isabel Gracia Luengo y Pedro García Sevilla. *Introducción a la programación con Python 3*. Universitat Jaume I, 2014. ISBN: 978-84-69711781.
- [4] Gregor Novak y Evelyn Patterson. “Just-in-Time Teaching. Across the Disciplines and Across the Academy”. En: Stylus Publishing, 2010. Cap. An Introduction to Just-in-Time Teaching (JiTT), págs. 3-23.
- [5] Alberto Prieto Espinosa, Beatriz Prieto Campos y Begoña del Pino Prieto. “Una experiencia de flipped classroom”. En: *XXII Jornadas de Enseñanza Universitaria de la Informática. JENUI 2016*. (Almería). Jul. de 2016, págs. 237-244. ISBN: 978-84-16642-30-4.
- [6] Anthony Robins, Janet Rountree y Nathan Rountree. “Learning and Teaching Programming: A Review and Discussion”. En: *Computer Science Education* 13.2 (2003), págs. 137-172. URL: <http://dx.doi.org/10.1076/csed.13.2.137.14200>.
- [7] Thomas J. Shuell. “Cognitive Conceptions of Learning”. En: *Review of Educational Research* 56.4 (1986), págs. 411-436.
- [8] Scott P. Simkins y Mark H. Maier, eds. *Just-in-Time Teaching. Across the Disciplines and Across the Academy*. New Pedagogies and Practices for Teaching in Higher Education series. Stylus Publishing, 2010. ISBN: 978-1-57922-293-2.