

Introducción de una herramienta OCL en la asignatura de Ingeniería de Requisitos

Loli Burgueño¹, Javier Luis Cánovas Izquierdo¹, Elena Planas¹, Jordi Cabot^{1,2}

¹Estudis d'Informàtica, Multimèdia i Telecomunicació, Universitat Oberta de Catalunya; ²ICREA
{lburguenoc, jcanovasi, eplanash}@uoc.edu, jordi.cabot@icrea.cat

Resumen

En muchas asignaturas de Ingeniería Informática se pide a los alumnos que programen en papel. Entre sus ventajas, esto permite a los alumnos centrarse en la funcionalidad evitando las distracciones causadas por la sintaxis y sin limitar su pensamiento a un lenguaje o paradigma de programación concreto. Aún así, los estudiantes se quejan de no poder probar su código, lo que les impide descubrir errores.

En contraposición al estado del arte que ha estudiado este problema para lenguajes de propósito general como Java con estudiantes de cursos iniciales, nosotros nos centramos en lenguajes estándares de la definición de reglas como OCL que se imparten en cursos avanzados.

En este artículo presentamos nuestra experiencia tras introducir una herramienta de modelado para la definición de restricciones OCL en una asignatura de Ingeniería de Requisitos. Esta asignatura es optativa y se ofrece en el tercer y cuarto curso del grado de Ingeniería Informática. El estudio que hemos realizado abarca dos cursos académicos, 2019 y 2020, en los cuales hubo 136 y 161 estudiantes matriculados respectivamente. Presentamos los detalles de nuestro experimento, los resultados obtenidos del estudio empírico llevado a cabo y nuestras conclusiones, las cuales respaldan la idoneidad del uso de herramientas.

Abstract

In many Computer Engineering courses, students are asked to program on paper. Among its advantages, this allows students to focus on functionality avoiding the distractions caused by syntax and without limiting their thinking to a specific programming language or paradigm. Still, students complain about not being able to test their code, preventing them from discovering bugs.

While the state of the art has studied this problem for general-purpose languages like Java and with students of initial courses, we focus on standard languages for constraint definition like OCL, which are taught in advanced courses.

In this article, we present our experience after introducing a modeling tool for the definition of OCL constraints in the Requirements Engineering course. This course is optional and is offered in the third and fourth years of the Computer Engineering degree. The study we have carried out covers two academic years, 2019 and 2020, in which there were 136 and 161 students enrolled, respectively. We present the details of our study, the results obtained from the empirical study carried out and our conclusions, which support the suitability of the use of tools.

Palabras clave

Ingeniería de requisitos, herramientas, OCL

1. Introducción

En los últimos años, los profesores responsables de la asignatura de Ingeniería de Requisitos en la Universitat Oberta de Catalunya hemos detectado que los estudiantes mostraban cierto descontento durante la realización de una práctica evaluable en la que les pedíamos que definieran en papel una serie de restricciones usando el lenguaje declarativo OCL [8]. Las peticiones de algunos estudiantes y la sospecha de que el desarrollo de este ejercicio en papel y sin ningún soporte digital pudiera estar ocasionando un problema de aprendizaje nos llevó a plantearnos la introducción de una herramienta de modelado con soporte para que permita la ejecución de dichas restricciones OCL.

El uso del papel en el ámbito de la programación tiene sus defensores y sus oponentes. Por una parte, muchos educadores respaldan sus beneficios para estudiantes de ingeniería informática o cualquier otra enseñanza técnica superior. Estos afirman que la programación en papel permite a los estudiantes centrarse en la lógica del programa que están escribiendo, evita las distracciones causadas por errores de sintaxis y no limita a los estudiantes a pensar en un lenguaje de programación o paradigma específico¹. Estas afirmaciones también

¹<https://classcube.com/write-code-paper/>

están respaldadas por muchas empresas del ámbito nacional e internacional, las cuales lo hacen explícito, por ejemplo, durante su proceso de contratación. Una de las varias pruebas que las empresas hacen a sus potenciales trabajadores suele ser un ejercicio en una pizarra, es decir, piden a sus entrevistados que resuelvan un problema en una pizarra usando el (pseudo-)lenguaje que ellos prefieran, y de este modo es cómo evalúan sus conocimientos y habilidades.

En las universidades, la programación con lápiz y papel se utiliza con frecuencia no solo como medio de aprendizaje sino también a la hora de evaluar los conocimientos de los estudiantes en ejercicios y prácticas evaluables y exámenes finales [1, 12, 13].

A pesar de sus beneficios y adopción, la programación en papel también tiene sus enemigos y algunos inconvenientes ampliamente reconocidos. Por ejemplo, los estudiantes suelen quejarse que no pueden ejecutar, probar y depurar el código que están escribiendo, lo que les impide verificar su corrección y detectar errores que podrían ser fácilmente descubiertos con una simple ejecución del programa.

Dado que no nos consta que haya ningún estudio que aborde el uso de papel en contraposición a una herramienta a la hora de aprender lenguajes formales y/o lenguajes estándares de la definición de reglas (como es el caso de OCL) y que sabemos que los lenguajes de modelado como UML/OCL son extensivamente usados, no solo en nuestra universidad sino en muchas otras², decidimos documentar y publicar nuestra experiencia.

Nuestro estudio muestra que el uso de herramientas para el aprendizaje de lenguajes de definición de reglas es percibido de forma positiva tanto por estudiantes como profesores. En comparación con los resultados del curso anterior, donde no se les ofreció a los estudiantes la posibilidad de usarla, las notas han mejorado.

El resto de este artículo está estructurado de la siguiente forma: el apartado 2 explica el contexto de la asignatura donde se ha realizado el experimento, el apartado 3 describe el experimento y la metodología que hemos seguido, el apartado 4 presenta los resultados obtenidos, en el apartado 5 presentamos nuestra discusión sobre los resultados, el apartado 6 el trabajo relacionado y, por último, el apartado 7 presenta brevemente nuestras conclusiones y líneas de trabajo futuro.

2. Contexto

El experimento que describe este artículo se contextualiza en la asignatura Ingeniería de requisitos del Grado en Ingeniería Informática de la Universitat Oberta de Catalunya (UOC). Se trata de una asignatura optativa que se imparte una vez al año y que los estudiantes pueden cursar a partir del tercer año. La asignatura parte de

un conocimiento básico sobre la ingeniería del software y profundiza en la primera etapa del ciclo de vida del desarrollo del software. Los contenidos de la asignatura se organizan en módulos docentes: (1) introducción a la ingeniería de requisitos; (2) obtención de requisitos; (3) gestión de requisitos; (4) documentación de requisitos; y (5) validación y verificación de requisitos. El lenguaje OCL se introduce en el módulo 4, como herramienta para documentar formalmente los requisitos.

El modelo de evaluación de la asignatura se basa en la evaluación continua, y está compuesta por cuatro pruebas (PEC) de carácter obligatorio. Todas las PEC trabajan un mismo caso de estudio a lo largo del semestre, lo que permite a los estudiantes tener una visión completa de todas las fases de la ingeniería de requisitos. En concreto: la PEC1 se centra en la obtención de los requisitos; la PEC2 trabaja la gestión de los requisitos de forma colaborativa en grupos de 4 o 5 personas; durante la PEC3 los estudiantes documentan los requisitos de forma ágil mediante casos de uso e historias de usuario; y, finalmente, en la PEC4, que es la actividad objeto de nuestro experimento, los estudiantes documentan los requisitos de modo formal (mediante UML/OCL) y aplican técnicas de validación y verificación. A modo de ejemplo, la PEC4 del semestre de 2020 consistía en documentar formalmente y validar los requisitos de una red social (ver Pregunta 14 en [6]). En concreto, dado un diagrama de clases, se pedía a documentar cuatro restricciones de integridad mediante OCL.

La evaluación de la asignatura es formativa, dado que se proporciona un retorno a los estudiantes de las diferentes PEC mediante una calificación acompañada de comentarios individuales, así como la publicación de la solución de cada una de ellas.

La asignatura cuenta con dos profesores responsables (los cuales son autores de este artículo) y tiene un alrededor de unos 150 estudiantes matriculados cada semestre de impartición, que se distribuyen en aulas virtuales de unos de 70 estudiantes cada una. Cada aula es dinamizada por un profesor colaborador, que guía y acompaña a los estudiantes durante su aprendizaje y finalmente los evalúa y les retorna el *feedback* de cada una de las actividades.

A lo largo del semestre, toda la comunicación se realiza de forma asíncrona y *online*, a través de los foros del aula virtual (donde los mensajes son públicos para todos los estudiantes del aula) y a través del correo electrónico (generalmente utilizado para resolver dudas directamente y de forma privada con los profesores).

3. Descripción del experimento

En este trabajo hemos construido un experimento con el objetivo de responder la siguiente pregunta de investigación: ¿Qué impacto tiene en el aprendizaje de los estudiantes el uso de una herramienta de modelado

²<https://mde-network.github.io/#miembros>

con soporte para la definición y ejecución de restricciones OCL? Proponemos medir el aprendizaje mediante tres medidas indirectas que nos permiten llevar a cabo una validación cruzada de los resultados: la autopercepción del estudiante, la percepción del profesorado y el rendimiento académico. Nuestra hipótesis es que el uso de la herramienta tiene un impacto positivo en la comprensión y aprendizaje del lenguaje OCL ya que la experimentación durante el proceso de aprendizaje les permite conocer la sintaxis y semántica del lenguaje, detectar errores tempranos (que pueden corregir y de los cuales aprender) y, por tanto, definir restricciones correctas con un mayor grado de éxito.

Dado nuestro contexto académico y el hecho de que estamos ante un caso de investigación-acción (o *Action Research* en inglés), para responder a la pregunta de investigación hemos hecho uso de los métodos empíricos más apropiados para abordar nuestra problemática [5].

Hemos construido un experimento controlado para estudiar la relación causa-efecto entre el uso de la herramienta de modelado (variable binaria independiente) y sus consecuencias. Hemos identificado un número de variables dependientes, entre ellas: tiempo dedicado, percepción de utilidad y dificultad de la herramienta, y rendimiento académico. A continuación presentaremos dichas variables más detalladamente.

El experimento se desarrolla en la cuarta PEC de la asignatura, que contiene tres ejercicios, donde uno de ellos evalúa los conocimientos sobre OCL y tiene un peso del 40%. En particular, el ejercicio de OCL presenta un diagrama de clases y contiene cuatro sub-apartados donde el estudiante debe definir restricciones OCL. En el semestre de desarrollo del experimento hemos continuado con la misma estructura para la PEC que en años anteriores y, para facilitar la comparación del impacto de utilizar una herramienta de modelado, las restricciones que deben definirse en los sub-apartados son similares a las requeridas en el mismo ejercicio de la PEC del semestre anterior (i.e., evalúan las mismas funcionalidades de OCL).

Antes de comenzar la actividad, los estudiantes son informados del experimento y se les ofrece la opción de utilizar una herramienta de modelado para resolver los ejercicios. La decisión final de utilizar dicha herramienta recae en el estudiante. La herramienta de modelado seleccionada fue MAGICDRAW, ya que es una de las opciones más populares para modelado en UML, soporta la definición y ejecución de OCL, y se utiliza en la asignatura de Ingeniería del Software, la cual es pre-requisito de nuestra asignatura.

Los estudiantes disponen de cuatro semanas para trabajar en la PEC y entregar su solución. Durante este tiempo, los estudiantes pueden publicar sus dudas en el foro del aula virtual o contactar directamente con el profesor colaborador de la asignatura vía email. Al

finalizar este periodo de tiempo, los autores de este artículo recopilamos los datos del experimento a partir de tres fuentes diferentes. Estas son: (1) los estudiantes, mediante el uso de un cuestionario online; (2) los profesores colaboradores, mediante entrevistas online; y (3) los medios de comunicación y calificaciones obtenidas. A continuación describimos en detalle cada parte.

3.1. Cuestionarios

En nuestro experimento recopilamos información del alumnado mediante cuestionarios anónimos online. La estructura y el contenido de los cuestionarios se definieron de forma iterativa por los autores de este trabajo. Una vez alcanzada la versión final, se validó con los profesores colaboradores.

Cada cuestionario tiene una primera sección con preguntas generales que nos permiten caracterizar al estudiante que participa en el experimento y conocer su conocimiento del lenguaje OCL. Para caracterizar al estudiante se incluyen preguntas demográficas típicas así como la nota del estudiante en la asignatura de Ingeniería del Software. Las preguntas sobre OCL están enfocadas a saber si el estudiante lo conoce, la percepción que tiene (i.e., si es un lenguaje fácil o difícil) y si ve necesario el uso de herramientas para practicarlo.

A continuación, la segunda sección del cuestionario contiene una serie de preguntas específicas de OCL que dependen de si el estudiante ha decidido utilizar MAGICDRAW o no. Para estudiantes que sí han utilizado la herramienta, el cuestionario incluye preguntas para conocer el motivo de dicha decisión, la experiencia del estudiante, el tiempo dedicado a resolver los ejercicios y la percepción de dificultad de cada sub-apartado de los ejercicios. Adicionalmente, se incluyen preguntas para conocer si MAGICDRAW ha permitido a los estudiantes experimentar más allá de los ejercicios, permitiendo explorar el lenguaje y resolver dudas de forma autónoma.

Si el estudiante no ha utilizado la herramienta de modelado, el cuestionario solamente incluye preguntas para conocer el motivo de dicha decisión, el tiempo dedicado a resolver los ejercicios y la percepción de dificultad para cada sub-apartado de los ejercicios.

El formulario completo puede encontrarse en [6].

3.2. Entrevistas estructuradas al profesorado colaborador

El proceso de recogida de comentarios del profesorado colaborador se planteó a través de entrevistas estructuradas. Estas entrevistas se organizaron para cubrir cuatro bloques: (1) impresión general, para conocer la percepción y opinión de los profesores sobre los efectos del uso de MAGICDRAW a la hora de resolver la PEC; (2) su opinión sobre si darle o no continuidad al uso de la herramienta y a la recomendación al alumnado de su

uso en cursos posteriores; (3) el grado de interacción con los estudiantes, para saber si el hecho de introducir la herramienta de modelado afecta esta dimensión (p.ej., conocer si tuvieron que responder muchas preguntas técnicas sobre el uso de MAGICDRAW no relacionadas con el lenguaje OCL); y (4) conclusión general y comentarios adicionales, para identificar, mejorar y proponer cambios en futuros semestres.

3.3. Medios de comunicación y rendimiento académico

Para complementar la evaluación de nuestro experimento, recopilamos el número de mensajes publicados en el foro y los emails referidos a los ejercicios de OCL así como las calificaciones de los estudiantes. De forma adicional, y con la intención de hacer un estudio comparativo, también recopilamos estos datos para el semestre anterior.

4. Evaluación empírica

4.1. Recolección de datos

Tal y como se comentó en la sección anterior, la recolección de datos se produjo al finalizar la PEC4 y los datos provienen de tres fuentes diferentes, para las cuales indicamos el momento de ejecución: (1) el cuestionario para los estudiantes se realizó durante la primera semana de enero de 2021; (2) las entrevistas al profesorado colaborador se realizaron a finales de enero de 2021 después de haber revisado y evaluado la PEC; y (3) los datos del registro de mensajes intercambiado en los foros del aula y las notas obtenidas en el curso 2020/2021 se recogieron después de las acciones anteriores.

4.2. Resultados

Encuesta a los estudiantes

En la encuesta hubo una participación del 32 % (52 de 161 estudiantes la respondieron). De ellos, 46 se identificaron con el género masculino, 6 con el femenino y no hubo estudiantes que se identificaran con otro. Un estudiante se encuentra en la franja de edad entre los 18-23 años, 41 estudiantes entre los 24-45 años y 10 estudiantes son mayores de 45 años.

En primer lugar, reportamos los resultados de la sección de preguntas generales. A la pregunta de si conocían OCL con anterioridad a este curso, el 75 % contestaron que no, el 19 % dijeron que habían oído hablar de él y solo el 6 % (3 estudiantes) dijeron que sí. Cabe destacar que para estos 3 estudiantes, esta era su primera matrícula en la asignatura, por lo que su uso ha debido producirse en otro contexto ajeno a nuestra asignatura. Cuando les preguntamos por su percepción general sobre OCL, en particular, sobre el nivel de dificultad de

	Núm. estud.	Moda	Mediana	Media	Std.
MH	2	-	0	0,50	0,71
Sobres.	3	0	0	0,33	0,57
Notable	27	0	0	0	0,88
Aprob.	7	0	0	-0,57	0,79
Conval.	13	0 y 1	0	-0,38	0,87
Total	52	0	0	-0,13	0,86

Cuadro 1: Percepción de la dificultad de OCL usando la escala de acuerdo de Likert [-2=muy complejo..+2=muy fácil] en función de la calificación obtenida en ingeniería del software.

OCL usando la escala de acuerdo de Likert de 5 valores que va desde muy complejo (-2) a muy fácil (+2), tanto la moda como la mediana de sus respuestas fue de 0.

A pesar del uso de la escala cuantitativa de Likert, la cual normalmente se traduce a valores enteros entre 1 y 5 a partir de los cuales no se suelen calcular medias, nosotros hemos optado por migrarla esta escala a una escala decimal en el rango [-2..2], y hemos calculado la media y desviación típica³.

En la encuesta preguntamos a los alumnos por la nota que obtuvieron en la asignatura de Ingeniería del Software (la cual es pre-requisito). A continuación, usando la misma escala anterior, evaluamos la opinión sobre OCL condicionada a la nota que dichos estudiantes obtuvieron en Ingeniería del Software y obtuvimos los resultados que mostramos en el Cuadro 1. Podemos observar una relación directa entre las notas de Ingeniería del Software y la percepción de los estudiantes: los estudiantes con notas más altas tienden a pensar que OCL es más fácil, mientras que los estudiantes con notas más bajas lo encontraron más complejo.

Cuando les preguntamos a los estudiantes si consideraban que el aprendizaje de OCL necesita de herramientas para poder hacer ensayo-error usando la escala de acuerdo de Likert. obtuvimos una moda de 2, una mediana de 1 y una media de 1,04 con una desviación típica de 1,03, lo cual indica que tienden a pensar que el uso de herramientas es beneficioso o incluso necesario.

El Cuadro 2 muestra los resultados estadísticos a esta misma pregunta considerando de forma independiente al grupo de 35 estudiantes que no usó MAGICDRAW (grupo de control) y al grupo de 52 estudiantes que sí lo usó (grupo experimental). La media de las opiniones muestra como incluso aquellos estudiantes que no usaron MAGICDRAW, tienden a pensar que su uso es más necesario que innecesario.

Respecto al tiempo dedicado a realizar el ejercicio de OCL, los estudiantes que usaron MAGICDRAW dijeron haber tardado una media de $6,11 \pm 4,61$ horas

³La UOC usa esta estrategia para obtener datos más precisos e informativos al analizar las encuestas de satisfacción del alumnado.

	Num. estud.	Moda	Media	Media	Std.
Sí MD	35	2	2	1,34	0,84
No MD	17	0	0	0,41	1,12
Total	52	2	1	1,04	1,03

Cuadro 2: Respuestas a la pregunta: “Considero que el aprendizaje de OCL necesita herramientas para poder hacer ensayo-error” usando la escala de acuerdo de Likert en función de si usaron MAGICDRAW (MD).

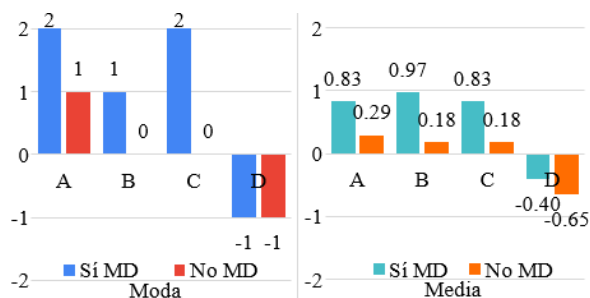


Figura 1: Percepción de los estudiantes sobre la dificultad del ejercicio de OCL.

mientras que los estudiantes que no lo usaron tardaron $5,56 \pm 4,84$ horas. Para comparar estadísticamente si hay diferencia en el tiempo que tardaron ambos grupos, hemos hecho la prueba t de Welch de dos colas y varianzas desiguales. Nuestra hipótesis nula es que los tiempos de los alumnos de ambos grupos no difieren. La prueba arroja un valor de 0,71, que al ser mayor que $\alpha = 0,05$, nos hace aceptar la hipótesis nula y concluir que no hay diferencias temporales significativas.

A continuación mostramos los resultados a las preguntas específicas a estudiantes que no usaron MAGICDRAW. Les preguntamos cuál había sido la razón para optar por no usar MAGICDRAW y 5 de ellos dijeron que “consideraban saber escribir restricciones y no necesitaban usar una herramienta”, 6 de ellos que “no consiguieron hacer MAGICDRAW funcionar”, 4 dijeron no haberlo usado “por no querer perder el tiempo usando una herramienta” y dos personas por “falta de tiempo”. Cabe destacar que varios estudiantes se quejaron de la incompatibilidad de MAGICDRAW con la nueva actualización de macOS. Curiosamente, de los 5 estudiantes que consideraban saber escribir OCL sin probarlo, 3 habían usado MAGICDRAW anteriormente, 2 pensaron usarlo pero luego descartaron la idea y uno de ellos dijo que la herramienta es difícil de usar y que no consideró que tuviera un beneficio sustancial.

Respecto a la percepción de dificultad para resolver cada uno de los subapartados del ejercicio de OCL (A-D), preguntamos a los estudiantes por el grado de dificultad mediante la pregunta: “Me ha resultado fácil resolver el apartado...”, y de nuevo usamos la escala de acuerdo de Likert y presentamos los resultados traslada-

dos al rango [-2..2]. La Figura 1 muestra a la izquierda en azul la moda de las respuestas que dieron los estudiantes que usaron MAGICDRAW y en rojo los que no lo usaron; y a la derecha, en azul turquesa la media de las respuestas que dieron los estudiantes que usaron MAGICDRAW y en naranja los que no lo usaron. Se puede observar cómo en general todos los estudiantes valoraron que los ejercicios A, B y C eran más fáciles que el ejercicio D. No obstante, se aprecia que quienes usaron MAGICDRAW consideraron que los ejercicios eran más fáciles.

Por último, presentamos los resultados a las preguntas específicas que hicimos a estudiantes que usaron MAGICDRAW. Usando nuevamente la escala de Likert, preguntamos a los estudiantes si:

- MAGICDRAW les había ayudado a resolver el ejercicio más rápidamente, para lo cual obtuvimos una respuesta de moda y media 0, lo cual indica que los estudiantes no consideraban que MAGICDRAW hubiera tenido ningún impacto temporal;
- MAGICDRAW les había permitido resolver el ejercicio más fácilmente, para lo cual obtuvimos una moda de 1 y una media de 0,54, que indica que los estudiantes tienden a estar de acuerdo;
- si MAGICDRAW les permite aprender OCL de manera práctica y plantear ejercicios más realistas y complejos, para la cual obtuvimos una moda de 0 y media de 0,63, lo cual indica que los estudiantes tienden a estar de acuerdo con esta afirmación;
- si recomendarían usar MAGICDRAW en cursos posteriores, a lo cual los estudiantes respondieron con una moda de 1 y una media de 0,86.

También preguntamos a los estudiantes si:

- consideraban que uso de MAGICDRAW les había permitido experimentar con restricciones OCL más allá de las que se pedían en los ejercicios; y 21 estudiantes (60 %) nos dijo que sí, mientras que 14 (40 %) dijeron que no;
- el uso de MAGICDRAW les había permitido resolver dudas de OCL de manera autónoma, sin tener que recurrir al profesor o a los foros del aula, y 23 estudiantes (66 %) nos dijeron que sí, 9 (26 %) dijeron que no, y un solo estudiante nos dijo que no le habían surgido dudas.

Con el objetivo de proporcionar claridad y permitir la reproducibilidad de nuestro análisis, tanto las preguntas del cuestionario como las respuestas de los estudiantes pueden ser consultadas en [6].

Entrevistas a los profesores colaboradores

Los tres profesores colaboradores, y especialmente dos de ellos que impartieron docencia en ambos cursos, coincidieron en que el cambio ha sido positivo y que

Curso	Núm. estud.	Msj. en foro	Msj. por email	Msj/ Estud.
2019	136	35	20	0,40
2020	161	61	70	0,81

Cuadro 3: Interacción profesorado-estudiantes

deberíamos seguir dando a los estudiantes la opción de usar una herramienta. También dijeron que no había habido muchas preguntas relacionadas con el uso de MAGICDRAW, más allá de algún problema de instalación en macOS BigSur.

Cuando les preguntamos si les había supuesto una mayor carga de trabajo, su respuesta fue negativa.

También nos dijeron que tenían la sensación de que las preguntas de los estudiantes este curso eran más concretas y que parecía que tenían más idea de cómo abordar los ejercicios, en contraposición a otros años en los que los estudiantes estaban más desorientados.

Como comentario adicional, uno de los profesores nos dijo que había notado que la calidad de las soluciones había aumentado y, sobre todo, le había llamado la atención que los estudiantes habían hecho un uso más correcto de la sintaxis del lenguaje. Apuntó a que en años anteriores algunos estudiantes tenían tendencia a escribir consultas con una sintaxis más propia de SQL que de OCL, mientras que este año ha encontrado que esto no ocurría con la misma frecuencia.

Comparativa curso 2019 y 2020

El Cuadro 3 recoge los resultados sobre la interacción entre los profesores y los estudiantes en foros y por email. En la última columna podemos ver como, para esta PEC, el ratio de mensajes por estudiante en 2019 era de 0,40 mientras que en 2020 incrementó a 0,81. Esto refleja que en 2020 hubo una mayor actividad en los foros del aula y más preguntas a los profesores.

El Cuadro 4 presenta los datos que nos permiten hacer una comparativa entre las calificaciones de los cursos 2019 y 2020. En ella hemos registrado el número de estudiantes, el número de esos estudiantes que no entregaron la PEC (columna NP) y la nota media y la desviación típica de la nota de la PEC4. Para probar si, estadísticamente, la nota en ambos cursos ha sido significativamente diferente, hemos realizado la prueba *t* de Welch con dos colas e iguales varianzas, siendo la hipótesis nula que no hay diferencia en las notas. La prueba arroja un valor de 0,0065, el cual al ser menor que $\alpha = 0,05$, nos hace rechazar la hipótesis nula y afirmar que las notas en ambos cursos han sido significativamente diferentes. Además, como puede darse el caso de que, por diversas razones y a pesar de haber usado PECs similares ambos años, haya diferencias en las notas, de cara a hacer una comparativa lo más justa posible, hemos recopilado la nota media final de la asignatura

Curso	Núm. estud.	NP	Nota PEC	Nota final	Nota PEC Norm.
2019	136	6	8,08 ± 1,14	7,91	6,39
2020	161	9	8,46 ± 1,15	7,92	6,70

Cuadro 4: Notas de la PEC en los cursos 2019 y 2020

natura en ambos cursos y hemos normalizado la nota de la PEC con respecto a ella. Así podemos concluir que mientras que la nota normalizada en el curso 2019 fue un 6,39; en el curso 2020 ha sido de 6,70, es decir, una media de 0,31 puntos más alta.

4.3. Amenazas a la validez

Por lo que respecta a la validez interna, los métodos de evaluación empíricos no son infalibles y diversos factores pueden influir en su validez, como por ejemplo, ignorar alguna variable que pueda tener un impacto en los resultados obtenidos y por tanto en las conclusiones. Es por ello que hemos intentado usar más de un método (cuestionario a los estudiantes, entrevistas estructuradas a los profesores y una comparativa cuantitativa de resultados objetivos entre ambos cursos) para intentar responder a la misma pregunta.

El cuestionario que proporcionamos a los estudiantes fue anónimo. Esto nos impidió hacer un análisis más profundo como por ejemplo relacionar el uso (o no uso) de MAGICDRAW con la nota final de los estudiantes. No obstante, creemos que nuestra decisión fue acertada ya que el anonimato mejora la honestidad de las respuestas.

Preguntar a los alumnos sobre si aconsejan el uso de MAGICDRAW tras haberlo usado o no puede propiciar un resultado sesgado. La validación cruzada mediante el uso de diferentes métodos empíricos contribuye a mitigar estos sesgos.

De los tres profesores colaboradores, dos de ellos formaron parte de la asignatura en ambos cursos y solo uno se incorporó en el año 2020. Esto garantiza que al menos estos dos profesores han evaluado e interactuado con los estudiantes de la misma forma. Además, están en posición de expresar su opinión personal sobre el desarrollo de ambos cursos.

Para poder hacer la comparativa entre los cursos 2019 y 2020 lo más justa posible, la PEC4 en 2020, aunque describiendo un caso de uso diferente, intentaba imitar fielmente la estructura y contenido de la PEC4 de 2019. Cabe decir que las PECs de años anteriores no se publican y, en teoría, los estudiantes no tienen acceso a ellas, con lo cual la similitud entre ambas no debería haber alterado las notas. Además, a la hora de comparar las notas de los cursos 2019 y 2020, hemos calculado la nota de la PEC normalizada para mitigar el impacto de la comparación PECs y estudiantes diferentes.

Por otro lado, la validez externa se puede ver amenazada ya que el experimento se desarrolla en un entorno

real donde los estudiantes deben tener la opción de elegir si usar MAGICDRAW o no, no pudimos hacer una asignación equitativa y aleatoria. Esto dio lugar a grupos no balanceados lo que puede afectar a los resultados estadísticos. No obstante, dado que el número de estudiantes en cada grupo es significativo, creemos que los resultados pueden ser generalizables a otros cursos y asignaturas donde se aborde la definición de restricciones OCL.

5. Discusión

En primer lugar, hemos observado que los estudiantes apoyan el uso de herramientas de modelado. Los resultados de nuestro experimento muestran que la percepción de dificultad de los estudiantes es significativamente menor cuando estos usan una herramienta de modelado. Los estudiantes que utilizaron MAGICDRAW reconocieron que la herramienta les permitió ser más autónomos y experimentar no solo con los ejercicios planteados en la PEC sino con más. Además, estos recomiendan el uso de MAGICDRAW en cursos posteriores.

El estudio del rendimiento académico de los estudiantes ha sido ligeramente más elevado. Aunque la encuesta a los estudiantes fue anónima y no podemos relacionar las respuestas con las notas de los estudiantes, la calificación final media ha sido ligeramente superior. Aunque esto apunta a un resultado satisfactorio, este aspecto requiere un estudio más profundo en los próximos cursos ya que, al no ser una diferencia excesivamente destacable, ésta podría estar ligada también a otros factores que no hayamos considerado en nuestro estudio.

El experimento ha mostrado que no existe una barrera de entrada técnica relevante. No hemos apreciado problemas significativos que dificulten la instalación y uso de MAGICDRAW más allá de los problemas de instalación en la última actualización del sistema operativo macOS. Cabe decir que MAGICDRAW no había tenido problemas en macOS con anterioridad, y que éste no ha sido el único software que falló tras la actualización a BigSur. Por lo tanto, lo consideramos más bien una desafortunada coincidencia que finalmente pudo ser resuelta mediante el uso de una máquina virtual.

Las entrevistas con los profesores colaboradores indican que estos apoyan la incorporación de herramientas de modelado. Ninguno apuntó a ningún inconveniente que las desaconsejara y todos estuvieron de acuerdo en que dar la opción a usar MAGICDRAW fue un cambio positivo, y recomiendan su uso de cara al futuro.

La introducción de herramientas de modelado no ha significado un aumento de dedicación del profesorado. Curiosamente, a pesar de que el número de mensajes intercambiados entre estudiantes y profesores durante el curso 2020 aumentó con respecto a 2019, en la entrevista a los profesores colaboradores, estos nos transmitieron que no habían notado una mayor carga de trabajo.

Esto probablemente es debido a que las preguntas de los estudiantes fueron más concretas y rápidas de resolver.

El uso de MAGICDRAW ha sido un complemento a las actividades y no ha implicado un rediseño de la evaluación de la asignatura. No obstante, si seguimos confirmando las ventajas del uso de herramientas, podremos plantearnos incorporar en el plan docente de la asignatura objetivos de aprendizaje más ambiciosos.

La introducción de herramientas supone a los profesores responsables el esfuerzo de hacer un estudio de todas aquellas que podrían ser usadas y, una vez se ha optado por una (o varias), la creación de manuales y actualización de los mismos año tras año. Esto es especialmente importante en el caso de MAGICDRAW ya que dependiendo de la versión de la herramienta y del tipo de licencia la funcionalidad es diferente.

6. Trabajo relacionado

Hay numerosos estudios en la literatura sobre el uso de herramientas de modelado. Muchos de estos estudios se basan en experimentos que comparan diversas características de las herramientas de desarrollo con el fin de determinar su usabilidad [11, ?]. Si nos centramos en OCL, Burgueño *et al.* [3] describen los principales problemas que se han encontrado al enseñar modelado en un curso dedicado a la Ingeniería del software, y presentan un caso de estudio con UML/OCL que ha sido utilizado con éxito durante el curso. No obstante, en este curso todos los estudiantes usaron la herramienta USE tanto para el modelado como para la definición de restricciones OCL y no se valoró las ventajas/desventajas de usar una herramienta frente a no usarla. Por otro lado, Maraee *et al.* [7] comparan la eficacia de desarrollar restricciones usando dos enfoques diferentes: el lenguaje declarativo OCL mediante la herramienta USE, y el lenguaje imperativo Java mediante un IDE de Java.

En menor medida, también existen estudios que comparan el uso de herramientas versus la escritura con lápiz y papel y su impacto en el aprendizaje de los estudiantes. Buena parte de estos estudios, que se centran principalmente en el aprendizaje de lenguajes de programación, determinan las ventajas de la evaluación electrónica por delante la evaluación tradicional (en papel u oral). Bessedsen *et al.* [2] reportan un experimento con exámenes de laboratorio en una asignatura introductoria de programación con más de 500 estudiantes. El resultado fue satisfactorio en todos los aspectos: simplicidad, eficacia, rendimiento y satisfacción de los participantes (docentes y estudiantes). Por otro lado, Rytönen *et al.* [10] describen su experiencia enseñando programación en C. Sus conclusiones determinan que los estudiantes perciben los exámenes de programación en formato electrónico como más realistas y naturales en comparación con exámenes en papel. En esta misma línea, Bottcher *et al.* [4] informan sobre su

transición del papel a los exámenes electrónicos para un curso de introducción a la programación con Java.

Finalmente, un estudio similar a nuestro trabajo es el publicado por Öqvist et al. [9], donde los autores describen su experimento para evaluar los efectos en el rendimiento de los estudiantes comparando la codificación a mano frente a herramientas de programación. En este caso, los resultados del experimento, en el que los sujetos eran estudiantes principiantes y se usó Java, no fueron concluyentes. Lamentablemente, no tenemos conocimiento de estudios similares centrados en el uso de herramientas de modelado frente a dibujos manuales para lenguajes gráficos como UML o lenguajes de definición de reglas como OCL.

7. Conclusiones

Este artículo presenta un estudio donde se compara el uso de una herramienta de modelado para definir restricciones OCL en contraposición a la definición de las mismas en papel. El estudio usa diferentes métodos empíricos (p.ej., entrevistas y cuestionarios) e involucra a diferentes sujetos (estudiantes y profesores colaboradores) para hacer una validación cruzada y así confirmar la validez de los resultados. El estudio concluye que los estudiantes tienen una percepción positiva hacia el uso de herramientas para definir restricciones OCL, y que los profesores colaboradores también aprueban y aconsejan su uso en cursos posteriores. Además, los resultados parecen indicar que el uso de MAGICDRAW tiene cierto impacto en la nota final de los estudiantes, siendo ligeramente más alta cuando estos lo usan.

Este estudio es un primer intento hacia mejorar el aprendizaje de lenguajes estándares de definición de reglas por parte de los estudiantes universitarios así como su satisfacción con el proceso de aprendizaje. Después de este experimento, hemos optado por mantener el uso opcional de MAGICDRAW en nuestra asignatura y seguiremos analizando la evolución del rendimiento y la satisfacción de los estudiantes. También estudiaremos qué otros factores (como por ejemplo los materiales proporcionados u otras herramientas de modelado) pueden ayudar a mejorar el rendimiento y la satisfacción de los estudiantes, así como soluciones para poder relacionar los cuestionarios con la nota obtenida por el estudiante en la actividad asegurando su anonimato. A largo plazo, el éxito en el uso de estas herramientas nos permitirá a los profesores plantearnos objetivos de aprendizaje más ambiciosos tanto en esta asignatura como su propagación a otras asignaturas de ingeniería del software.

Referencias

- [1] Jony Adkins y Diana Linville: *Testing Frequency in an Introductory Computer Programming Course*. Information Systems Education Journal, 15:22–28, 2017.
- [2] Jens Bennedsen y Michael E. Caspersen: *Assessing Process and Product – A Practical Lab Exam for an Introductory Programming Course*. En *Proc. of Frontiers in Education (FIE)*, 2007.
- [3] Loli Burgueño, Antonio Vallecillo y Martin Gollá: *Teaching UML and OCL models and their validation to software engineering students: an experience report*. Computer Science Education, 28(1):23–41, 2018.
- [4] Alex Böttcher, Veronika Thurner y Daniela Zehetmeier: *Alignment of Teaching and Electronic Exams and Empirical Classification of Errors for an Introductory Programming Class*. En *Proc. of CSEET'20*, páginas 1–10, 2020.
- [5] Steve Easterbrook, Janice Singer, Margaret-Anne D. Storey y Daniela E. Damian: *Selecting Empirical Methods for Software Engineering Research*. En *Guide to Advanced Empirical Software Engineering*, páginas 285–311. 2008.
- [6] Loli Burgueño, Javier Luiz Cánovas Izquierdo, Elena Planas, and Jordi Cabot: *Materiales de MagicDraw/OCL (curso 2020)*. <http://hdl.handle.net/20.500.12004/1/C/JENUI/2021/001>.
- [7] Azzam Maraee, Eliran Nachmani y Arnon Sturm: *Constraints Specification Via Tool Support: A Controlled Experiment*. J. Object Technol., 19(3):3:1–18, 2020.
- [8] Object Management Group: *Object Constraint Language (OCL) Specification. Version 2.4*, 2014.
- [9] Martina Öqvist y J. Nouri: *Coding by hand or on the computer? Evaluating the effect of assessment mode on performance of students learning programming*. Journal of Computers in Education, 5:199–219, 2018.
- [10] Anni Rytönen y Venla Virtakoivu: *Comparative Student Experiences on Electronic Examining in a Programming Course - Case C*. En *Proc. of Koli Calling*, 2019, ISBN 9781450377157.
- [11] Safdar Aqeel Safdar, Muhammad Zohaib Iqbal y Muhammad Uzair Khan: *Empirical Evaluation of UML Modeling Tools-A Controlled Experiment*. En *Proc. of ECMFA'15*, páginas 33–44, 2015.
- [12] Michael de Raadt Anne Philpott Judy Sheard Mikko Jussi Laakso Daryl D'Souza James Skene Angela Carbone Tony Clear Raymond Lister Geoff Warburton Simon, Donald Chinn: *Introductory Programming: Examining the Exams*. En *Proc. of ACE'12*, volumen 123, 2012.
- [13] Arto Vihavainen, Matti Paksula y Matti Luukkainen: *Extreme Apprenticeship Method in Teaching Programming for Beginners*. En *Proc. of SIGCSE '11*, página 93–98, 2011, ISBN 9781450305006.