

La Experimentación en la docencia de Ingeniería del Software

Félix García, Manuel Serrano, José A. Cruz-Lemus, Marcela Genero, Coral Calero y
Mario Piattini

Grupo Alarcos
Escuela Superior de Informática
Universidad de Castilla-La Mancha
13001 Ciudad Real

e-mail: {Felix.Garcia, Manuel.Serrano, JoseAntonio.Cruz, Marcela.Genero, Coral.Calero, Mario.Piattini}@uclm.es

Resumen

Los estudios empíricos en Ingeniería del Software son fundamentales para la validación de diversos métodos, técnicas, herramientas, etc., y los alumnos juegan un papel fundamental a la hora de llevarlos a cabo. Estos estudios no permiten obtener beneficios centrados exclusivamente en los aspectos de investigación, sino que es muy importante considerar también sus beneficios en la docencia. En este artículo se estudia la aplicación de experimentos controlados en cursos de Ingeniería de Software, destacando los beneficios que estos estudios aportan a los alumnos y a los docentes e investigadores que los llevan a cabo. Además, se presentan los resultados obtenidos en la realización de tres experimentos en cursos de ingeniería del software. Como consecuencia de llevar a cabo estos experimentos se han obtenido importantes beneficios pedagógicos.

1. Introducción

Uno de los problemas de la Ingeniería del Software consiste en que a menudo se proponen una gran diversidad de métodos, lenguajes, herramientas, entornos, etc., de los que no se demuestra su utilidad práctica. Por ello, los métodos empíricos han centrado la atención de la comunidad científica en la ingeniería del software durante los últimos años. Mediante los métodos empíricos es posible evaluar nuevas aportaciones en la ingeniería del software antes de que sean introducidas en los procesos software de las empresas [10].

Los estudios empíricos más comúnmente utilizados en la Ingeniería del Software son: experimentos controlados, casos de estudio y encuestas, los cuales difieren fundamentalmente

en sus objetivos y restricciones. En el entorno académico los estudios empíricos más significativos tanto desde el punto de vista investigador como docente son los experimentos [1].

A la hora de llevar a cabo experimentos, los alumnos juegan un papel muy importante, ya que antes de realizar estos estudios en entornos industriales (lo que requiere un gasto significativo de tiempo, esfuerzo y recursos) en muchas ocasiones los investigadores llevan a cabo estudios piloto con alumnos en entornos académicos [4]. De hecho, hay que considerar que los alumnos constituyen la próxima generación de profesionales [12]. Por ello, los resultados de estos estudios en entornos académicos tienen una gran importancia y permiten obtener conclusiones significativas de cara a su realización posterior en entornos industriales. Bajo ciertas circunstancias, las diferencias entre los alumnos y los profesionales son pequeñas y las tareas requeridas en ciertos experimentos no requieren experiencia industrial, por ello se puede considerar la experimentación con alumnos como viable [9], [2].

Además, los estudios empíricos no sólo constituyen una aportación científica, sino que también proporcionan importantes beneficios pedagógicos en cursos de Ingeniería del Software, por lo que se establece una importante conexión entre la investigación y la docencia que es fundamental analizar [4].

Hoy en día los alumnos de cursos de Ingeniería del Software empiezan a participar en experimentos (sobre todo en universidades norteamericanas y británicas). Como resultado de ello una gran cantidad de métodos y técnicas han podido ser validadas empíricamente tal y como se refleja en numerosas publicaciones científicas. Sin embargo, muchas de estas publicaciones se

centran en presentar los beneficios que los estudios han aportado a la investigación, dejando de lado a los alumnos, lo que incluso hace pensar en ocasiones, que los alumnos han sido “utilizados” de forma egoísta para obtener resultados en la investigación. Por ello es fundamental abordar los beneficios que la experimentación aporta desde el punto de vista pedagógico y aportar estos beneficios a los alumnos cuando se planifican y se llevan a cabo experimentos en entornos académicos.

El principal objetivo de este trabajo es el de estudiar la aplicación de experimentos en cursos de Ingeniería de Software, destacando los beneficios que estos estudios aportan, tanto desde el punto de vista pedagógico como investigador.

A continuación, en el apartado 2 se identifican los distintos beneficiarios de los experimentos y las ventajas que estos estudios les aportan, sobre todo para el investigador, el docente y los alumnos. En el apartado 3 se describen las fases que hay que considerar a la hora de llevar a experimentos. Los resultados de diversos experimentos llevados a cabo con alumnos en distintos cursos de Ingeniería de Software, se presentan en el apartado 4. Finalmente se presentan las conclusiones obtenidas y las consideraciones para el futuro.

2. Beneficiarios de la Experimentación en Ingeniería del software.

Cuando se desarrollan experimentos se pueden identificar cuatro beneficiarios principales de los resultados. Cada uno de los cuales tendrán diferentes puntos de vista [4]:

- **Investigador.** Es el encargado de planificar y llevar a cabo el experimento. Su objetivo es demostrar la utilidad práctica de su propuesta u obtener conclusiones preliminares para llevar a cabo experimentos en entornos industriales.
- **Docente.** Es el profesor responsable de la asignatura o grupo de alumnos, que constituyen el contexto en el que se lleva a cabo el experimento. Su principal objetivo es enseñar los conocimientos y habilidades relacionadas con los estudios empíricos realizados y que van a ser de utilidad a los alumnos en su trabajo como futuros profesionales.

- **Alumnos.** Son los sujetos utilizados en el experimento. Su objetivo es aprender técnicas y habilidades que les puedan servir como futuros profesionales.
- **Empresa.** Las empresas de software son las beneficiarias últimas de los estudios empíricos. Como resultado de dichos estudios, las empresas pueden adoptar nuevos métodos o tecnologías que influirán en la mejora de sus procesos software y en definitiva les permitirán obtener beneficios económicos a medio o largo plazo.

En el contexto académico es de especial relevancia establecer las principales ventajas de los estudios empíricos desde el punto de vista del docente e investigador, que en muchas ocasiones son la misma persona, y sobre todo el beneficio que se aporta a los alumnos. En la Tabla 1 se resumen las ventajas más significativas de llevar a cabo estudios empíricos con alumnos de acuerdo al análisis realizado en [4]:

Beneficiario	Beneficios
Docente	<ul style="list-style-type: none"> • Nueva forma de formar a los alumnos respecto de la enseñanza tradicional, alumno • Fomento de la participación en grupo de los alumnos en determinados estudios empíricos • Mejora de la comunicación con los alumnos • Nuevas formas de evaluación de los alumnos en situaciones en las que no tienen el estrés típico de un examen formal • Introducir la ingeniería del software empírica como parte de la enseñanza en la ingeniería del software
Investigador	<ul style="list-style-type: none"> • Evidencia preliminar para aceptar o rechazar hipótesis • Demostrar a las empresas software la relevancia de la investigación y la utilidad de llevar a cabo estudios empíricos en las propias empresas • Prever los recursos necesarios para realizar experimentos en entornos industriales y preparar el material necesario del experimento para realizarlo en la industria • Formación de investigadores noveles en el desarrollo de estudios

Beneficiario	Beneficios
	empíricos
Alumno	<ul style="list-style-type: none"> • Formación en materias complementarias a la formación de grado • Conciencia de nuevos problemas a resolver en el software en general y en las industrias en particular • Mejor autoevaluación de cuál es su nivel en determinados temas de ingeniería del software que en las clases tradicionales más centradas en los aspectos teóricos • Percepción de las ventajas de usar métodos empíricos en la ingeniería del software • Preparación para su futuro profesional en el que en muchas ocasiones serán sometidos a cuestionarios, informes, encuestas, etc.

Tabla 1. Beneficios de los estudios empíricos en entornos académicos desde la perspectiva del docente, del investigador y de los alumnos

En la práctica, es muy importante conocer los efectos negativos de llevar a cabo experimentos en entornos académicos, con el fin de que la planificación de los experimentos sea suficientemente cuidadosa para que estos efectos no se produzcan [4]:

- Desde el punto de vista del **investigador** los inconvenientes de realizar estudios empíricos con alumnos son el esfuerzo necesario para prepararlos, ya que nunca deben olvidarse de los aspectos pedagógicos al preparar el estudio. Otro problema son las amenazas a la validez del estudio, tema que se aborda brevemente en el apartado 3.
- Desde el punto de vista del **docente** los problemas que pueden surgir son la necesidad de motivar al docente para llevar a cabo el experimento, ya que le supone un esfuerzo de formación mucho mayor para esa clase que si se tratara de una clase normal y además debe motivar a los propios alumnos creando el ambiente necesario en la clase. El docente debe tener capacidad de atender cualquier duda de los alumnos. Estos problemas se solucionan en gran medida si el docente y el investigador son la misma persona, situación

que es deseable a la hora de realizar estudios empíricos con alumnos.

- Desde el punto de vista de los **alumnos** pueden surgir problemas derivados de pérdida de tiempo, si se trata de experimentos que requieren una formación extensa de los alumnos que les hace perder varias clases en lugar de aprovecharlas formándoles en temas más interesantes o útiles para su futuro profesional. Otro problema potencial es que el resultado del experimento demuestre que la técnica o método que acaban de aprender no es efectivo, aunque en este caso también se puede encontrar la parte pedagógica, consistente en demostrarles que una tecnología o técnica no se puede aceptar sin evaluarla en la práctica por muy nueva que sea. Además, en este último caso se pueden intentar hallar las causas del resultado.

En definitiva, es fundamental considerar las ventajas e inconvenientes comentados en la planificación de estudios empíricos, con el fin de que las ventajas sean percibidas por los alumnos y se puedan evitar en la medida de lo posible los problemas que de ello se puedan derivar, como la desmotivación o el descontento. Los autores de este artículo han llevado a cabo como docentes/investigadores experimentos con alumnos en diferentes cursos de la Escuela Superior de Informática de Ciudad Real. Los resultados de dichos experimentos centrados en los aspectos más pedagógicos se describen en el apartado 4. A continuación se describen de forma resumida las principales características y aspectos a considerar a la hora de llevar a cabo experimentos controlados con alumnos.

3. Proceso Experimental

En este apartado se describen los factores esenciales que hay que considerar a la hora de llevar a cabo una buena planificación y realización de experimentos controlados, con el fin de obtener resultados que sean creíbles [11],[17],[13], [3], [12].

Las ventajas de los experimentos es que pueden determinar las situaciones en las que ciertas afirmaciones son verdaderas y pueden proporcionar el contexto en el que ciertos estándares, métodos y herramientas son

recomendables. Sólo si el experimento se realiza adecuadamente, seremos capaces de sacar conclusiones acerca de las relaciones entre la causa y el efecto para la cual formulamos la hipótesis (la cual queremos corroborar mediante la realización de experimentos) Los experimentos necesitan ser planeados cuidadosamente si queremos que nos proporcionen resultados útiles y significativos. Por ello es necesario seguir un proceso experimental como el que se propone en [17] que consta de las siguientes etapas:

1. **Definición.** Este es el primer paso, en él definiremos el experimento en términos del problema y los objetivos.
2. **Planificación.** Donde determinaremos el diseño del experimento y consideraremos la instrumentación del mismo.
3. **Operación.** Esta es la fase en la que se lleva a cabo el experimento y se recogen los datos empíricos.
4. **Análisis e interpretación.** Los datos recogidos son analizados utilizando técnicas estadísticas. Además se interpretan los resultados obtenidos.
5. **Evaluación de la validez.** En este paso evaluaremos los aspectos que pueden amenazar a la validez del experimento Para ello se consideran se estudian cuatro tipos de validez:
 - **Validez de constructo.** Define hasta qué punto las medidas utilizadas para medir las variables independientes y dependientes, miden fielmente los conceptos que intentan medir.
 - **Validez interna.** Se refiere al grado de confiabilidad con el que podemos asegurar la relación causa-efecto entre las variables independientes y dependientes.
 - **Validez externa.** Consiste en el poder de generalización de los resultados, no sólo para el entorno donde se ha desarrollado el experimento sino también en otros entornos.
 - **Validez de la conclusión.** Define hasta qué punto las conclusiones obtenidas son estadísticamente válidas.
6. **Presentación y difusión.** En esta última fase se elabora un informe sobre los resultados para que los mismos sean difundidos con el objetivo de otros investigadores puedan y replicar el experimento.

4. Experimentación con alumnos en cursos de Ingeniería del Software.

En este apartado se presentan los resultados de tres experimentos controlados llevados a cabo en asignaturas de Ingeniería del Software. Los experimentos tuvieron lugar en horario normal de clase, su realización era voluntaria y se motivó especialmente a los estudiantes para su realización, destacando los beneficios que los experimentos les proporcionan como futuros profesionales. Para conseguir evitar las posibles amenazas a la validez de los experimentos, tomamos una serie de medidas:

- Los sujetos que se convocaron para hacer cada experimento tenían una experiencia y unos conocimientos parecidos.
- Los dominios de los diagramas eran lo suficientemente sencillos y comunes para que no existiese ningún problema a la hora de entenderlos.
- Para evitar los efectos de aprendizaje los esquemas fueron entregados a cada sujeto en un orden diferente.
- Los sujetos que realizaron el experimento era la primera vez que realizaban un experimento de este tipo, por lo que los efectos de la persistencia están atenuados.
- Los sujetos estaban motivados pues los ejercicios formaban parte de los conocimientos que debían adquirir en su formación.
- No se permitió que los sujetos hablaran entre ellos durante la prueba ni que pudieran copiar los resultados unos de otros.
- Todas las dudas fueron resueltas por la persona que supervisaba el experimento.
- Los sujetos no tenían conocimiento, a priori, de los aspectos que se pretendían estudiar ni cuales eran las hipótesis que se habían planteado.

4.1. Experimento sobre Métodos de Diseño de Almacenes de Datos

Definición

Se han propuesto diversos métodos de diseño de almacenes de datos basados en los diagramas de estrella, ya que estos diagramas aumentan la

eficacia y la comprensión de los esquemas de los almacenes de datos. Aunque esta afirmación es ampliamente aceptada, no se ha demostrado empíricamente que sea cierta, por lo consideramos adecuado realizar un experimento para poder comprobarlo.

El objetivo de este experimento [14] era determinar si los diseños de almacenes de datos basados en diagramas de estrella son más comprensibles que aquellos realizados basándose en el diseño tradicional de bases de datos.

Planificación

El experimento se realizó con 11 alumnos de la Escuela Superior de Informática de Ciudad Real (UCLM). Todos nuestros sujetos tenían conocimientos de diseño y uso de bases de datos, además poseían conocimientos del diseño y uso de los almacenes de datos pues habían recibido esta información como parte de sus estudios de doctorado.

El experimento constaba de tres diagramas diseñados según el diseño tradicional de bases de datos y otros tres diagramas semánticamente equivalentes a los anteriores usando diagramas en estrella, sobre los que había que realizar unas consultas utilizando SQL. Los diagramas representaban problemas reales y sus dominios y eran lo suficientemente sencillos como para ser comprendidos fácilmente.

El objetivo del experimento era analizar si había diferencia en los tiempos de respuesta entre los diagramas semánticamente equivalentes para analizar si alguno resultaba más complejo.

Operación

Antes de realizar el experimento se explicó el tipo de problemas que debían resolver, cómo se debía contestar a las preguntas y que material se estaba proporcionando para su realización.

Análisis e Interpretación

Según los objetivos del estudio y la configuración del experimento usamos una prueba ANOVA de medidas repetidas univariante y como resultado de este análisis estadístico se llegó a la conclusión de que no hay diferencia en la comprensión de los esquemas debido al método de diseño utilizado.

También se dedujo que el dominio de los esquemas tampoco influye en la comprensión de los mismos.

Conclusiones

Como conclusión del experimento podríamos deducir que no hay diferencia en la comprensión de los esquemas debido al método de diseño utilizado y que por tanto da igual diseñar un almacén de datos utilizando una metodología tradicional o los diagramas en estrella.

Cuando comunicamos los resultados a los sujetos, intentamos sacar algunas conclusiones de estos resultados. No parecían adecuados los resultados que obteníamos pues el modelado en estrella es ampliamente utilizado, lo que permitió a los alumnos dudar de los prejuicios y las ideas ampliamente aceptadas y no demostradas.

No obstante, intentamos ver cuales eran las causas que podían haber afectado a los resultados y obtuvimos las siguientes:

- El tamaño de los esquemas del experimento no era muy grande.
- Los dominios eran lo suficientemente conocidos como para no tener repercusión en la complejidad y la comprensión.
- Los sujetos no estaban tan acostumbrados a trabajar con diagramas en estrella y esto puede afectar al tiempo de realización de las consultas.

Los beneficios obtenidos de la realización de este experimento fueron varios, como que los alumnos se vean dentro de una investigación empírica y sepan como realizar estos experimentos. Además los alumnos aprendieron a dudar de los prejuicios y a analizar las causas por las que los resultados obtenidos no fueron los esperados.

4.2. Experimento de Modelos de Procesos Software

Definición

El modelado y la medición de los procesos software se han convertido en dos elementos fundamentales para promover la mejora de calidad de los mismos. Con el fin de evaluar la complejidad estructural de los modelos de

procesos software se han definido un conjunto representativo de métricas [6]. Para validar las métricas propuestas se llevó a cabo un experimento cuyo objetivo fue establecer la relación entre la complejidad estructural de los modelos de procesos y su mantenibilidad a través de dos de sus subcaracterísticas: la entendibilidad y la modificabilidad. La mantenibilidad de un modelo de procesos es la facilidad con la que puede ser mantenido. Así, modelos de procesos software difícilmente mantenibles, pueden afectar a la ejecución de los proyectos (duración, coste, etc.) y a la calidad final de los productos obtenidos.

Planificación

El experimento se realizó con 45 alumnos de ingeniería técnica en informática de gestión y 41 de ingeniería técnica en informática de sistemas de tercer curso de la Escuela Superior de Informática de Ciudad Real. Todos los sujetos tenían conocimientos de modelado del producto pero no del modelado del proceso software. Por ello se les impartió una clase en la que se les explicó la importancia de la gestión efectiva de los procesos software para mejorar la calidad de los productos finales, haciendo especial hincapié en la influencia que en ello tiene el modelado del proceso. Como lenguaje de modelado del proceso software se explicó el lenguaje SPEM [16].

El material del experimento constaba de diez modelos de procesos software desarrollados a partir de diversos estándares y metodologías existentes en la bibliografía. El objetivo del experimento consistía en evaluar si la complejidad de los modelos de procesos (variable independiente) influye en la entendibilidad y en la modificabilidad de los modelos (variables dependientes).

Los objetos usados fueron diez modelos de procesos. Cada modelo incluía una hoja de ejercicios compuesta por una sección que podía consistir en cinco cuestiones relacionadas con los modelos de procesos o en cuatro ejercicios consistentes en realizar modificaciones sobre los modelos. En total cada sujeto tenía cinco modelos con cuestiones relacionadas con la entendibilidad y cinco modelos con cuestiones relacionadas con la modificabilidad.

La variable independiente fue medida mediante las métricas propuestas, mientras que las

variables dependientes fueron medidas calculando el tiempo que los sujetos tardaron en responder las cuestiones relacionadas con la entendibilidad de cada modelo (tiempo de entendibilidad) y mediante el tiempo que los sujetos emplearon en realizar las modificaciones que se requerían para cada modelo (tiempo de modificabilidad) respectivamente. Nuestra suposición es que cuánto menos tiempo se emplea en entender y modificar un modelo de procesos, más fácil será mantenerlo.

Operación

En primer lugar se impartió una sesión preparativa previa en la que se resolvió un ejemplo similar al material que luego se les entregaría.

Análisis e Interpretación

Utilizando el coeficiente de correlación de Spearman se analizó si existía correlación entre las métricas para la complejidad estructural de los modelos de procesos software y los tiempos de entendibilidad y de modificabilidad de los mismos. Como resultado se obtuvo que varias de las métricas estaban correlacionadas con dichos tiempos, por lo que consideramos que –en cierta manera– son válidas.

Conclusiones

Desde el punto de vista del investigador se llegó a la conclusión de que las métricas definidas eran – en cierta manera– válidas, y podrían ser utilizadas como indicadores del tiempo de entendimiento y mantenimiento de los modelos de procesos software, además de servir para poder comparar modelos semánticamente equivalentes.

Desde el punto de vista del alumno, la realización del experimento en la asignatura de Ingeniería del Software de tercer curso, permitió impartir en dicha asignatura un tema relacionado con el proceso software. En dicho tema se estudió la definición del Proceso Software, sus elementos más importantes, la importancia de la gestión de su calidad y se profundizó en el modelado de los mismos. Ello permitió a los alumnos tener una visión más amplia de la asignatura de Ingeniería del Software, ya que el proceso software sólo se estudia al tratar los modelos de ciclo de vida del software, y durante el curso aplican técnicas del

modelado del producto, pero no del proceso. Por ello consideramos que el experimento les beneficia como futuros profesionales.

Además, también hay que considerar los beneficios pedagógicos que podemos obtener cuando se les comunican los resultados. Para ello, se imparte una clase en la que por un lado se les hace ver la importancia de realizar estudios empíricos y por otro, se les motiva comparando sus resultados con los resultados del mismo experimento llevado a cabo en un entorno industrial, que han sido similares [7]. Ello demuestra que en determinadas circunstancias no existen diferencias significativas entre los alumnos de últimos cursos de carrera y los profesionales.

4.3. Experimento de Diagramas de Clases UML

Definición

Con la idea de que cuánto más complejos sean los diagramas de clases UML más difícil será entenderlos y luego mantenerlos se definieron un conjunto de métricas para medir la complejidad estructural y tamaño de tales diagramas [8]. Para validar empíricamente dichas métricas se llevó a cabo un experimento controlado. Por ello, el principal objetivo de este experimento era averiguar si realmente existe correlación entre la complejidad estructural y el tamaño de los diagramas de clases UML y la entendibilidad y mantenibilidad de los mismos.

Planificación

Los sujetos que realizaron el experimento fueron 24 alumnos del tercer curso de la Ingeniería Informática, en la Escuela Superior de Informática de la Universidad de Castilla-La Mancha, en Ciudad Real. Dichos alumnos estaban cursando el primer curso de Ingeniería de Software de la carrera. Por lo que no los podemos considerar expertos, aunque también habían cursado otras asignaturas de programación orientada a objetos en cursos previos.

Las variables independientes eran la complejidad estructural y el tamaño de los diagramas de clases UML, medidas a través de un conjunto de métricas definidas en [8].

Las variables dependientes eran la entendibilidad y la mantenibilidad de los diagramas de clases UML, medidas a través del tiempo que los sujetos emplearon para realizar las tareas requeridas en el experimento.

El experimento consistió en 9 diagramas de clases UML sobre distinto dominio de aplicación, pero lo suficientemente general para ser fácilmente comprendido por los sujetos. Las tareas a realizar fueron de dos tipos:

- Contestar el cuestionario adjunto a cada diagrama de clases, que consistía en cinco preguntas, y anotar el tiempo que tardaban en contestarlas, al que llamamos tiempo de entendimiento (expresado en segundos).
- Modificar cada diagrama de clases de acuerdo a cuatro nuevos requerimientos y anotar el tiempo utilizado, al que llamamos tiempo de mantenimiento (expresado en segundos).

Se eligió un diseño experimental intrasujetos, lo que significa que todos los sujetos debían realizar todas las mismas tareas sobre los 9 diagramas de clases UML.

Operación

Antes de realizar el experimento el profesor que supervisaba el experimento realizó una sesión de entrenamiento, en la que se les entregó a los sujetos un diagrama de clases con una serie de preguntas y tareas de mantenimiento, similares a las que luego deberían realizar en el experimento.

Se les comentó a los alumnos que esto les serviría para adquirir práctica en el diseño de diagramas de clases de UML, ya que no sólo es importante saber diseñarlos completamente, sino también saber entenderlos y poder modificarlos.

Como resultado obtuvimos los datos del cálculo de las métricas de la complejidad estructural y el tamaño de cada diagrama y los tiempos de entendibilidad y modificación que cada sujeto ha necesitado para realizar las tareas requeridas sobre cada diagrama. Adicionalmente se han comprobado las tareas realizadas para establecer unos índices de corrección y completitud.

Análisis e Interpretación

Analizando los datos recogidos mediante el coeficiente de correlación de Spearman obtuvimos que la mayoría de las métricas definidas estaban correlacionadas con el tiempo de entendimiento y mantenimiento de los diagramas de clases UML.

Conclusiones

Desde el punto de vista del investigador se llegó a la conclusión de que las métricas definidas eran – en cierta manera– válidas, y podrían ser utilizadas como indicadores del tiempo de entendimiento y mantenimiento de los diagramas de clases UML. Además de servir para poder comparar diagramas semánticamente equivalentes.

Desde el punto de vista del alumno la realización del experimento les permitió obtener cierta experiencia en el entendimiento y modificación de diagramas de clases UML, tareas que podrían tener que realizar tanto en los exámenes como en el desempeño de la profesión.

En una clase posterior, se les explicó a los alumnos el objetivo de investigación que perseguía el experimento. Se les explicaron brevemente las métricas definidas, las hipótesis planteadas y los resultados obtenidos. Por lo que creemos, que además pudieron aprender un tema que aún no habían visto en Ingeniería de Software que fue la utilización de métricas.

5. Conclusiones

En este artículo se ha abordado la realización de experimentos en entornos académicos, destacando las ventajas que estos estudios aportan tanto desde el punto de vista del docente/investigador como del punto de vista del alumno. Además, se han presentado los resultados obtenidos en la realización de tres experimentos en cursos de ingeniería del software. Estos resultados confirman los beneficios docentes obtenidos, destacando los siguientes:

- Formación en los principios básicos de la experimentación software.
- Aprendizaje de los procesos software, su modelado y tecnología.
- Importancia del uso de métricas para evaluar cuantitativamente técnicas, métodos y estándares en Ingeniería del Software.
- Evaluación de distintas técnicas de diseño.

- Práctica en el diseño de diagramas de clases con UML.
- Realimentación a partir de los resultados de los experimentos.

El entusiasmo mostrado por la mayoría de los alumnos al realizar el experimento y su interés por conocer los resultados obtenidos nos lleva a pensar que este tipo de experimentos son muy beneficiosos y se deberían realizar siempre que sea posible en cursos de Ingeniería del Software, tal y como ya se está realizando en otros centros de estudio internacionales [5], [10].

No obstante, a la hora de llevar a cabo experimentos en cursos de ingeniería del software siempre hay que considerar los beneficios que se pueden aportar a los alumnos y además hay que tener en cuenta diversas consideraciones éticas [15], para que como resultado del experimento no se produzca desmotivación y descontento por parte de los alumnos.

Referencias

- [1] Baresi, L., Morasca, S. and Paolini, P. Estimating the Design Effort of web Applications. *Proceedings of the 9th International Software Metrics Symposium (METRICS'03)*, pp. 62-71, 2003.
- [2] Basili V., Shull F. and Lanubile F. Building Knowledge Through Families of Experiments. *IEEE Transactions on Software Engineering*, Vol. 25 N° 4, pp. 435-437, 1999.
- [3] Briand, L., Arisholm, S., Counsell, F., Houdek, F. and Thévenod-Fosse, P. (2000). Empirical Studies of Object-Oriented Artefacts, Methods, and Processes: State of the Art and Future Directions. *Empirical Software Engineering*, Vol. 4 N° 4, pp. 387-404, 2000.
- [4] Carver, J., Jaccheri, L., Morasca, S. y Shull, F. Issues in Using Students in Empirical Studies in Software Engineering Education. *Proceedings of the 9th International Software Metrics Symposium (METRICS'03)*, pp. 239-251, 2003.
- [5] Carver, J., Jaccheri, L., Morasca, S. y Shull, F. Using Empirical Studies during Software Courses. *Experimental Software Engineering Research Network 2001-2003*. LNCS 2765, pp. 81-103, 2003.

- [6] García, F., Ruiz, F. y Piattini, M. "Medición del Proceso Software", *VIII Jornadas de Ingeniería del Software y Bases de Datos*, Actas de las Jornadas E. Pimentel, N. Brisaboa, J. Gómez (eds), Alicante (España), pp. 303-314, 2003.
- [7] García, F., Ruiz, F. y Piattini, M. Definition and Empirical Validation of Metrics for Software Process Models. *5th International Conference on Product Focused Software Process Improvement (PROFES'2004)*. Lecture Notes in Computer Science (LNCS 3009), pp. 146-158, 2004.
- [8] Genero, M. Defining and Validating Metrics for Conceptual Models. Ph.D. Thesis Department of Computer Science, University of Castilla-La Mancha, 2002.
- [9] Höst, M., Regnell, B. and Wholin, C. Using Students as Subjects – A comparative Study of Students & Professionals in Lead-Time Impact Assessment". *4th Conference on Empirical Assessment & Evaluation in Software Engineering (EASE)*, Keele University, UK, pp. 201-214, 2000.
- [10] Höst, M. Introducing Empirical Software Engineering Methods in Education. *Proceedings of the 15th Conference in Software Education and Training (CSEET'02)*, pp. 170-179, 2002.
- [11] Juristo, N. and Moreno, A. *Basics of Software Engineering Experimentation*. Kluwer Academic Publishers, 2001.
- [12] Kitchenham, B, Pfleeger, S., Pickard, L., Jones, P., Hoaglin, D., El Emam, K. and Rosenberg, J. Preliminary Guidelines for Empirical Research in Software Engineering. *IEEE Transactions on Software Engineering*, Vol. 28 N° 8, pp. 721-734, 2002.
- [13] Perry, D., Porter, A. and Votta, L. Empirical Studies of Software Engineering: A Roadmap, *Future of Software Engineering*, ACM, Ed. Anthony Finkelstein, 2000, pp. 345-355.
- [14] Serrano, M., Calero, C. and Piattini, M. An Empirical Study with Datawarehouse Design Methods, *1st International Workshop Empirical Studies in Software Engineering*, Bunse, C., Jedlitschka, A. (eds), pp. 49-54, Finland, 2002.
- [15] Singer, J. and Vinson, N. Ethical Issues in Empirical Studies of Software Engineering. *IEEE Transactions on Software Engineering*, Vol. 28 N° 12, pp. 1171-1180, 2002.
- [16] Software Process Engineering Metamodel Specification; adopted specification, version 1.0. Object Management Group. November (2002). Available in <http://cgi.omg.org/cgi-bin/doc?ptc/02-05-03>.
- [17] Wohlin C., Runeson P., Höst M., Ohlson M., Regnell B. and Wesslén A. *Experimentation in Software Engineering: An Introduction*. Kluwer Academic Publishers, 2000.