

Aprendizaje experiencial e introducción de estresores en el diseño de una práctica centrada en procesos de gestión

Alejandro Catala*, Nelly Condori-Fernandez**, José A. Taboada*, José M. Cotos*, Julián Flores*

*Centro Singular de Investigación en Tecnologías Inteligentes, Universidade de Santiago de Compostela

**Departamento de Ciencias de la Computación, Universidade de A Coruña

{alejandro.catala, joseangel.taboada, manel.cotos, julian.flores}@usc.es,
n.condori.fernandez@udc.es

Resumen

La formación de los futuros ingenieros de software requiere de un equilibrio entre el desarrollo de habilidades técnicas y blandas (i.e., *soft-skills*). Este trabajo explora el diseño de una sesión práctica que, basada en principios de *aprendizaje experiencial* de Kolb, persigue que aprendan de la reflexión y experiencia durante la realización de un mini-proyecto en equipo bajo condiciones de estrés. Entre los procesos propuestos en la práctica están el control de la calidad o la gestión de cambios que no son intuitivos para los alumnos y se abandonan rápidamente bajo presión. El diseño introduce estresores de manera controlada para forzar errores de los que los alumnos puedan aprender. Con esto se persigue que los estudiantes sean más conscientes de la relevancia de las habilidades no-técnicas y de gestión.

Abstract

Soft-skills must be fostered over and above technical skills in order to ensure balanced and resourceful future software engineers. This work explores the design of a workshop session, which based on *experiential learning* principles seeks learning from own experience by reflecting and iterating on a team project. In the designed session, some processes such as quality control or changes management are especially promoted since they are not typically intuitive to students and are abandoned under pressure. The design introduces stressors in a controlled and timed manner in order to cause forced mistakes to trigger learning. This pursues that students become more aware of the relevance of managerial and non-technical skills.

Palabras clave

Ingeniería Software, diseño práctica, procesos de ingeniería, habilidades blandas, estrés, trabajo en equipo.

1. Introducción

Las *habilidades blandas* (i.e., *soft skills*) son una serie de capacidades relacionadas con el trabajo en equipo, la comunicación, el análisis, gestión de tiempo y la auto-organización entre otras, cuyo desarrollo resulta de especial relevancia para la industria del software [3]. Es por ello que los Grados en Informática, más allá del desarrollo de capacidades técnicas, deben también preparar a su alumnado para esta demanda del mercado laboral, y por tanto las metodologías educativas deben también evolucionar como se sugiere en [9] para asegurar un desarrollo profesional completo de sus egresados incorporando estas habilidades.

Las asignaturas de Ingeniería de Software (IS) son particularmente interesantes en este papel porque muchas de las metodologías, técnicas y herramientas presentadas [8, 10, 11] requieren de trabajo en equipo con problemas complejos, ya que es ahí donde alcanzan su máximo sentido y utilidad.

Por esta razón, se ha incorporado a la programación docente de la asignatura, una sesión práctica que obligue a los alumnos a definir y ejecutar procesos como Gestión de la configuración o Control de cambios que requieren poner en práctica las habilidades mencionadas. Se pretende de esta forma verificar si el alumnado ha alcanzado una comprensión del proceso que le permita llevarlo a la práctica y en qué grado entiende su importancia y valor para el éxito de los proyectos.

Dada la idoneidad de incorporar metodologías activas en las sesiones prácticas que permitan a los alumnos aprender de la experiencia [6] a través de aproximaciones basadas en proyectos, la contribución de este trabajo se centra en el diseño de una sesión práctica que fundamenta la experiencia docente a llevar a cabo, y esperamos sirva como recurso docente que inspire sesiones con similares necesidades.

2. Diseño de la sesión

2.1. Contexto didáctico

La asignatura de Ingeniería del Software (tercer curso del grado de Ingeniería Informática de la Universidad de Santiago de Compostela) presenta el desarrollo de software como un elemento de ingeniería que se rige por la aplicación de principios y buenas prácticas. Se presentan los modelos clásicos de ciclos de vida [10], y algunas normas y marcos para el desarrollo de proyectos [1, 11], visibilizando la importancia de los procesos y su mejora continua. Como aproximación práctica a los procesos fundamentales de la IS se diseña un taller que propone la realización de un trabajo bajo condiciones de estrés y presión. Introduciendo una fuerte restricción temporal se logra que el alumnado abandone procesos que no comprende, provocando los errores típicos de su no realización: repetición de trabajo, debido a su pérdida o a su realización simultánea por varios miembros del equipo; cambios inconclusos o inconsistentes; falta de calidad evidentes en las entregas etc. Se visualizan así los errores más comunes al trabajar en grupo y bajo presión. En esta fase el alumnado ya maneja el concepto de planificación y el uso de MSProject. Este taller podría tener cabida en cualquier otra materia relacionada, como Gestión de Proyectos Informáticos. El primer objetivo de aprendizaje del taller es hacer conscientes a los alumnos de la dificultad que tiene: a) definir los requisitos de un producto en un documento, b) crear un plan para desarrollar dichos requisitos, c) mantener la calidad y la consistencia entre los documentos cuando se realizan cambios bajo presión y d) organizar eficazmente el trabajo en grupo de forma que cada miembro sepa lo que debe hacer en cada momento, en función de su rol. Como segundo objetivo se persigue que, a través del análisis de las dificultades encontradas, el alumnado comprenda el valor de los procesos relacionados con la especificación, gestión de la configuración, calidad, planificación y control de cambios.

2.2. Fundamentos

Entre las teorías de aprendizaje cognitivo activo, el *aprendizaje experiencial* [6] destaca por ir un paso más allá del aprendizaje por acción. Teoriza que para incorporar conocimiento basado en la propia experiencia, otros procesos necesarios son de reflexión activa, conceptualización abstracta y experimentación activa como ilustra la Figura 1. Bajo este marco de trabajo, entendemos el taller como una oportunidad para que el alumnado experimente y aprenda de la reflexión sobre la experiencia, adquiriendo conocimiento no factual. Cada una de las fases en las que se estructura la sesión (véase Sección 2.3) supone una unidad de tra-

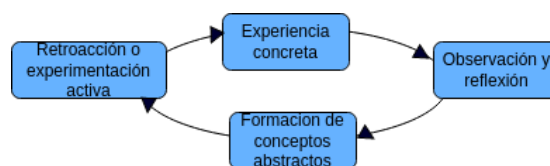


Figura 1: Proceso del aprendizaje experiencial [6].

bajo sobre la que los estudiantes tienen la oportunidad de tomar sus propias decisiones operativas, reflexionar sobre sus observaciones, redefinir las estrategias y conceptos puestos en práctica y proponer cambios en el proyecto en curso. En conjunto, estas fases conducen a la compleción del trabajo en equipo.

Con el fin de reforzar que el aprendizaje pueda darse a partir de la experiencia y comisión de errores, decidimos utilizar desencadenadores emocionales¹. En este trabajo consideramos varios de los tipos de desencadenadores (estresores) identificados por Kanjo et al. [5]: (i) ambientales (sonido de alarma cada vez que culmina el tiempo asignado a una fase); (ii) interacción con otros (profesor que introduce cada fase, observadores tomando nota del progreso de los grupos de trabajo); y (iii) fracasos, que pudieran producirse ya sea por la falta de tiempo o por desconocimiento para cumplir ciertas tareas adicionales que pueden ser solicitadas en alguna determinada fase del proyecto.

2.3. Guión de la sesión

El taller tiene una duración de 2,5 horas a lo largo de una única sesión práctica², pero las lecciones aprendidas derivadas son referidas por los docentes durante el resto de asignatura. Los estudiantes trabajan en equipo a la vez que, intencionadamente y de manera guionizada, se les expone a situaciones cambiantes en su proyecto, que genera sobrecarga de trabajo y/o conflictos que pueden afectar seriamente al desempeño y calidad de los resultados. Por conveniencia, y dado el tiempo disponible limitado, la sesión sigue una aproximación del flujo de procesos de gestión del tiempo propuesta en la tercera edición del PMBOK[4] para desarrollar el plan de un proyecto con las fases siguientes:

1. Inicio (Kick-off)
2. Definición del alcance
3. Definición de las actividades
4. Establecimiento de la secuencia y dependencias de las actividades
5. Revisión del plan
6. Estimación de recursos
7. Resolver sobreasignaciones. Redistribuir
8. Cierre (Entrega)

¹Estimulo externo o interno que provoca emociones positivas o negativas [5].

²Recurso: <https://enso.pages.citius.usc.es>

Cada una de estas fases comienza con la explicación de la tarea a realizar que se cierra con una transparencia, visible durante el resto de la fase, en la que se muestran las salidas que se esperan de la ejecución correcta de la fase. Tras esta explicación, el grupo dispondrá de 15 minutos de tiempo de trabajo efectivo para cada una de las fases, con la única excepción de la fase 1 de inicio, que contará con 10 minutos. Cada una de estas fases y tareas se les explican e imponen al alumno sin una justificación explícita ya que se espera que el aprendizaje del alumno se centre en los procesos de establecimiento de requisitos, admitir y gestionar los cambios y gestionar la calidad del documento cuya definición y realización depende de ellos.

En esta primera fase, se les detalla el contenido y estructura del documento que deben desarrollar y mantener, los procesos que deben llevar a cabo y los roles considerados con sus responsabilidades: Jefe de Proyecto (modera y dirige las discusiones y tareas, control de tiempos), Gestor documental (implementa y coordina toda la documentación), Gestor de cambios (toma nota de los cambios solicitados, asegura su realización y evita los cambios sin control), Asegurador de Calidad (advierde errores en los procesos, asegura que los documentos reflejan con claridad toda la información), Validador de requisitos (asegura que todas las actividades sirvan para cumplir algún requisito, y que todos los requisitos sean atendidos y se redacten sin ambigüedad).

En la fase 2, definición del alcance, se les describe el proyecto a realizar, que consiste en “transformar dos despachos contiguos de profesorado en un aula de informática con el mismo equipamiento del aula en la que están”, lo que asegura un completo conocimiento del dominio. A pesar de que podría parecer lógico proponer un proyecto de desarrollo de software, nuestra experiencia nos muestra que los alumnos, en este punto, no son conscientes de lo que implica realmente la realización de un proyecto de software completo, que en cambio debería resultar obvio en un proyecto de remodelación como el propuesto. Aún así, es frecuente que la presión les lleve a olvidar cosas tan evidentes como comprar los equipos informáticos o cablear el local.

En las fases 3 y 4, al trabajo asignado, se le añaden tareas específicas, que actúan como estresores, obligándoles a modificar los documentos que están creando manteniendo su consistencia. Además, no se les permite realizar los cambios de forma inmediata sino sólo su registro para implementarlos en una fase posterior, evitando la posibilidad de cambios al vuelo.

De esta forma en la fase 3, mientras intentan establecer las actividades necesarias para realizar el trabajo, se les solicita que añadan nuevos requisitos para forzar el uso de un proceso de control de cambios. Además,

se saca del aula a los alumnos con el rol “gestor de cambios” y se les alecciona para que no permitan cambios sin registrar en los documentos. La conversación se oculta al resto del equipo, lo que supone un estresor adicional de interacción. En la fase 4, cuando intentan organizar las actividades en un plan, se les añaden nuevas actividades (que no suelen considerar) y, para forzar el papel del asegurador de la calidad y aumentar la presión sobre los cambios, se contacta únicamente a los alumnos con este rol y se les entrega una *checklist* con aspectos que deberían estar presentes en el documento final de la entrega como índices, cabeceras o títulos numerados. La fase 5 es una fase utilizada para darles tiempo a completar tareas pendientes y en la 6, aunque se añade la tarea de seleccionar y añadir recursos, no se impone ninguna restricción sobre su cantidad, simplificando el trabajo. En la fase 7, en cambio, se limitan fuertemente los recursos apareciendo sobreasignaciones, que deben resolver para completar el trabajo y entregar de forma inmediata en la fase de cierre posterior.

2.4. Materiales e instrumentación

La sesión requiere de una aula de prácticas de laboratorio informático convencional que permita la distribución en grupos, además de un ordenador por alumno con acceso a internet y al menos un procesador de textos y un software para la planificación de proyectos. El aula debe contar con un proyector para la presentación de los resultados esperados para cada una de las fases. Hemos instrumentado la sesión con algunos cuestionarios en línea para obtener más información sobre el estrés y los principales aspectos relacionados del trabajo en equipo. Los cuestionarios se responden antes, durante e inmediatamente después de la sesión práctica. En particular el *Perceived Stress Scale* (PSS) [2] consiste en una escala autoadministrada utilizada típicamente como *gold-standard* en psicometría para valorar el estrés referido al último mes. Se administra antes de la fase inicial. Para medidas puntuales del estrés, se utiliza el *Stress Visual Analogue Scale* (Stress VAS) [7], que consiste de una regla que reporta entre 0 y 100 el nivel de estrés que siente el sujeto, asignando 0 a la etiqueta ‘Nada’ y 100 a ‘Tan malo como podría ser’. La escala implementada (ver figura 2) se administra en 5 instantes diferentes, inmediatamente después de las fases 1 (t_0), 3 (t_1), 4 (t_2), 5 (t_3) y 7 (t_4). Esta distribución obedece a la introducción de estresores en dichas fases, forzando la comisión de errores y magnificando carencias en las decisiones de gestión del trabajo en equipo. Esto permitirá sondear cómo evoluciona el estrés a lo largo de la sesión, y explorar cómo éste influye en sus decisiones y percepciones. Por último, los cuestionarios contemplan la inclusión de información de-

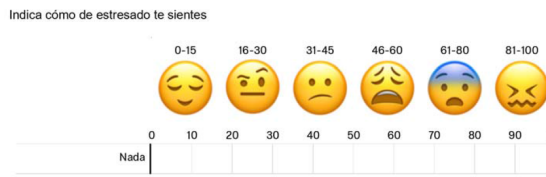


Figura 2: Escala *Stress* VAS implementada.

mográfica y el trabajo en equipo ³, que se administran en la fase de cierre, una vez concluida la entrega. Este cuestionario se utiliza como material de reflexión para el alumno, de forma que al responderlos analizan lo experimentando durante la sesión. Sus respuestas, analizadas globalmente, se utilizarán en la próxima clase para enfatizar las lecciones aprendidas durante el taller.

3. Conclusiones y trabajo futuro

El guión diseñado en este recurso docente recoge la estructura para conducir una única sesión práctica que permita la realización de un proyecto asumible por los estudiantes pero con la suficiente potencia como para forzar sus errores que evidencien problemas típicos en procesos de gestión cuando estos son llevados bajo presión. La contribución del presente diseño se resume con: 1) la incorporación de ideas sobre el aprendizaje experiencial de Kolb; 2) utilización de estresores a lo largo de un proyecto a desarrollar en una sesión de práctica; e 3) incorporar medición de estrés por medio de cuestionarios, tales como *PSS* y *Stress VAS*.

Como parte del trabajo futuro más inmediato se procederá a implantar y validar en la aula la práctica diseñada, mediante la recolección y análisis de las respuestas de los estudiantes, que hasta el momento se ha probado en una fase piloto. Se examinará cómo es la evolución del estrés a lo largo de la sesión, y si hay una relación entre el rol del individuo y los métodos de gestión definidos por cada equipo. Asimismo, se comprobará cómo los estudiantes reflexionan sobre los procesos en el trabajo en equipo, y se valorará su impacto contrastando con otros estudiantes que no realicen esta práctica. Esto nos permitirá proponer mejoras en relación a intervenciones experimentales futuras sobre la relación de la tolerancia al estrés basado en rasgos de personalidad con los roles a desempeñar, de tal forma que se favorezca el desempeño en equipo y desarrollo de habilidades.

³Basado en Schreyer Institute for Teaching Excellence de la Penn State University: https://www.schreyerinstitute.psu.edu/pdf/Team_Peer_Evaluation_Examples.pdf

Agradecimientos

Agradecimientos a la beca *MICINN-Juan de la Cierva* (IJC2018-037522-I), al proyecto *KUSISQA* (014-2019-FONDECYT-BM-INC.INV), a Consellería de Cultura, Educación e Ordenación Universitaria (grant accreditation ED431G-2019/04) y FEDER.

Referencias

- [1] Mary Beth Chrissis. *CMMI: Guidelines for Process Integration and Product Improvement (2nd Edition)*. Addison-Wesley Professional, nov 2006.
- [2] Sheldon Cohen. *Perceived stress in a probability sample of the United States.*, pages 31–67. The Claremont Symposium on Applied Social Psychology. Sage Publications, Inc, Thousand Oaks, CA, US, 1988.
- [3] Maya Daneva, Andrea Herrmann, Nelly Condori-Fernandez, y Chong Wang. Understanding the most in-demand soft skills in requirements engineering practice: Insights from two focus groups. En *Proceedings of the Evaluation and Assessment on Software Engineering, EASE '19*, page 284–290, New York, NY, USA, 2019. ACM.
- [4] Project Management Institute. *A Guide to the Project Management Body of Knowledge, Third Edition (PMBOK Guides)*. Project Management Inst, nov 2004.
- [5] Eiman Kanjo, Luluah Al-Husain, y Alan Chamberlain. Emotions in context: examining pervasive affective sensing systems, applications, and analyses. *Personal and Ubiquitous Computing*, 19:1197–1212, 2015.
- [6] David A Kolb. *Experiential learning: Experience as the source of learning and development*. FT press, 2014.
- [7] F.-X. Lesage, S. Berjot, y F. Deschamps. Clinical stress assessment using a visual analogue scale. *Occupational Medicine*, 62(8):600–605, 09 2012.
- [8] Roger S. Pressman. *Ingeniería de Software (Spanish Edition)*. McGraw-Hill Interamericana Editores S.A. de C.V., mar 2010.
- [9] C. R. Rupakheti, M. Hays, S. Mohan, S. Chenoweth, y A. Stouder. On a pursuit for perfecting an undergraduate requirements engineering course. En *2017 IEEE 30th Conference on Software Engineering Education and Training (CSEE T)*, pages 97–106, Nov 2017.
- [10] Ian Sommerville. *Software Engineering (9th Edition)*. Pearson, mar 2010.
- [11] Mario G. Piattini Velthuis. *Análisis y diseño detallado de aplicaciones informáticas de gestión*. Ra-Ma S.A. Editorial y Publicaciones, sep 2003.