

Estrategias Didácticas para el Aprendizaje y la Enseñanza del Pensamiento Computacional en el Nivel Académico Universitario

Natalia Colussi¹ y Natalia Monjelat²

¹ Licenciada, Facultad de Ciencias Exactas, Ingeniería y Agrimensura, UNR, Rosario, Argentina

² PhD, Instituto Rosario de Investigaciones en Ciencias de la Educación (IRICE, CONICET-UNR), Rosario, Argentina
colussi@fceia.unr.edu.ar, monjelat@irice-conicet.gov.ar

Resumen. El siguiente artículo presenta los avances de una línea de investigación en desarrollo que busca contribuir a mejorar el proceso de enseñanza y aprendizaje de la programación en carreras afines a las Ciencias de la Computación (CC). Puntualmente, se propone alcanzar este objetivo a partir de una revisión de las estrategias didácticas, empleando la Metodología de Enseñanza Basada en Proyectos (ABP), las actividades grupales, y las estrategias de resolución de problemas basadas en el Pensamiento Computacional. De esta forma, se han realizado numerosos cambios e innovaciones en el desarrollo y puesta en obra del cursado. A partir de las experiencias implementadas, es posible señalar múltiples aprendizajes y fortalezas de este enfoque tales como la posibilidad de seguimiento del progreso en el aprendizaje, la rápida adaptación al cambio de contexto presencial-virtual, la disminución del plagio al tratarse de producciones originales y creativas, entre otras. Estos resultados ponen en valor la estrategia del ABP para la enseñanza de conceptos de CC, particularmente en este caso, desde contenidos disciplinares de programación y el pensamiento computacional y las técnicas de resolución de problemas.

Keywords: Primer Curso de Programación en la Universidad, Aprendizaje Basado en Proyectos y Problemas, Pensamiento Computacional.

1 Introducción

Desde hace ya varios años, tanto Argentina como en el resto del mundo, existen miles de puestos de trabajo sin cubrir en el área de las Ciencias de la Computación, por la falta de recursos humanos con la formación y los conocimientos necesarios. Se observa a su vez, que las carreras afines a éstas ciencias presentan en el último tiempo un ingreso masivo, pero al mismo tiempo una gran deserción en los primeros años [1,2]. Según diversos autores, en esta problemática intervienen varios factores. Entre ellos se señalan al desconocimiento inicial de las disciplinas, la falta de adaptación al

nivel universitario, así como también las dificultades propias de las materias disciplinares impartidas en los primeros años.

A partir del reconocimiento de estas problemáticas, y particularmente de la última, la línea de investigación que aquí se presenta tiene por objetivo mejorar el proceso de enseñanza y aprendizaje de la programación en carreras afines a las Ciencias de la Computación a través de la Metodología de Enseñanza Basada en Proyectos, las actividades grupales, y las estrategias de resolución de problemas basadas en el Pensamiento Computacional. El contexto en el que se desarrolla la propuesta, es el segundo dictado de las materias de programación de los primeros años de las carreras Licenciatura en Ciencias de la Computación (LCC), Licenciatura en Matemática (LM), y Profesorado en Matemática (PM).

La propuesta del redictado que se propone desde la línea de investigación, intenta recurrir a estrategias de enseñanza y aprendizaje que se han mostrado eficaces para la construcción de saberes disciplinares en diferentes contextos y áreas disciplinares [3,4]. De esta forma, el ABP aparece como una opción innovadora para la enseñanza de la programación, haciendo del redictado una experiencia diferente al dictado inicial [5]. Asimismo, los proyectos están sustentados en las premisas del pensamiento computacional [10] y las estrategias de resolución de problemas de Polya [11] adaptadas por Thompson [12] para el desarrollo de programas, estableciendo pilares disciplinares sólidos para el futuro en la carrera académica y profesional de los estudiantes

Para estudiar el impacto de la propuesta se han diseñado encuestas que se administran al finalizar cada cursada. Asimismo se cuenta con registros de actas de cursado y de regularización, que permiten observar la cantidad de estudiantes que inician y finalizan el curso. En este documento se presentarán algunos resultados parciales de estos análisis, considerando la extensión del artículo.

Las acciones desarrolladas se enmarcan en un proyecto de investigación radicado en la Facultad de Ciencias Exactas, Ingeniería y Agrimensura (FCEIA) dependiente de la Universidad Nacional de Rosario (UNR). La investigación conforma un proyecto bienal, extendido por pandemia al año 2022 resolución CS 436/2022 y que dispone como fecha de finalización 31/12/2022.

2 Motivación

En la cátedra de Programación I y Programación se imparten conocimientos referidos al Pensamiento Computacional, haciendo fundamental hincapié en los principios de programación [6, 7]. La adquisición y construcción del conocimiento disciplinar para los estudiantes de primer año de las carreras LCC, LM y PM no se produce para todos al mismo tiempo. Los estudiantes provienen de diferentes especialidades y contextos en su nivel medio, conviviendo así, en un mismo espacio áulico, alumnos con experiencias, intereses y expectativas dispares.

En los últimos años las cátedras de programación, en el primer cursado, presentan ingresos promedios superiores a los trescientos inscriptos, y heterogeneidad de saberes y niveles entre los cursantes. Los registros internos de actas de examen y planillas de regularización muestra que una gran cantidad de estudiantes, aproximadamente un 50% en promedio, no logran aprobar o regularizar este primer

curso de programación. De estos estudiantes libres, aproximadamente un 60% abandona la carrera, pero el 40% que persevera asiste al segundo dictado que se ofrece al siguiente cuatrimestre [15]. Este nuevo dictado, conocido en FCEIA como redictado, representa una segunda oportunidad para incorporar habilidades y/o competencias no alcanzadas en la primera instancia y es esta población de estudiantes sobre la cual se trabaja con el ABP desde el año 2017.

Para llevar adelante la propuesta basada en el ABP en la cátedra del redictado se desarrolló un “Plan de Trabajo Didáctico para el Aula” (PTDA) [8] redefiniendo el proceso de enseñanza-aprendizaje utilizado hasta el momento para brindar un mejor marco de contención y solución a los problemas que presentaba el estudiantado del curso. Esta iniciativa requirió de la reelaboración de la presentación, ejercitación y evaluación de contenidos propios de la materia. Se realizaron dos proyectos grupales de programación a lo largo del cuatrimestre los cuales permitieron a los estudiantes ejercitar y poner en práctica temas disciplinares, y ser evaluados así por los docentes a lo largo de todo el proceso de desarrollo de los mismos.

El trabajo de esta investigación se encuentra centrado entorno a los siguientes aspectos o lineamientos:

- Identificar detalladamente los problemas que manifiestan tener los estudiantes durante el primer curso de programación tanto en la faceta disciplinar como vincular con el medio académico.
- Implementar y adaptar las estrategias de enseñanza y aprendizaje activas, grupales, y colaborativas dentro de las cátedras Programación I y Programación.
- Desarrollar e investigar sobre distintas estrategias didácticas conjuntas y combinadas para abordar las problemáticas técnicas vinculadas al aprendizaje de la programación, como así también, a la inclusión de todos los estudiantes en su heterogeneidad en el proceso.
- Desarrollar e integrar actividades que fomenten aspectos motivacionales en los estudiantes, las cuales brinden el acercamiento entre pares dentro de la carrera, reconocimiento dentro de la comunidad académica, y el ímpetu de continuar y progresar dentro de la carrera; brindando así herramientas que perduren en el tiempo y sean fundacionales para mantener el proceso de aprendizaje y así la permanencia y el avance dentro de la universidad.
- Trabajar con experiencias que permitan desarrollar las denominadas *soft-skills* [14] (habilidades o competencias no medibles o difíciles de medir, de transmisión fundada en al experiencia) técnicas o disciplinares, como por ejemplo, entender procesos de refinamiento y mejora de diseño de funciones, resolución de problemas, como aquellas *soft-skill* vinculadas a aspectos más profesionales como por el ejemplo las comunicacionales e interpersonales, desarrollo de la creatividad, y la capacidad de imaginar soluciones a problemas originales, desarrollo de capacidad crítica y objetiva ante otras soluciones, valorar respuestas disímiles a la propia, etc.

3 Resultados

Uno de los resultados que nos ha dejado la experiencia es la importancia de acompañar el desarrollo de los proyectos de diferentes formas. La implementación del ABP en los primeros años de la universidad, permite observar claramente cómo los estudiantes progresan en la adquisición de conocimiento y saberes disciplinares, ya que conlleva un seguimiento mayor por parte de los docentes. Este acercamiento provoca que los docentes y estudiantes logren vincularse de una forma más próxima, haciendo que el seguimiento del proyecto por el cual se canaliza el aprendizaje, se obtiene una observación mucho más clara de cómo se logra la comprensión de los temas a través del diseño de la solución (modelo+estrategia+código) que los estudiantes producen como soluciones parciales de un proyecto.

El trabajo grupal es un pilar fundamental en la concreción de los proyectos. Esta forma de trabajo sostiene el trabajo y motiva a finalizarlo, brindando apoyo y contención entre pares tanto en la adquisición conjunta de saberes y competencias como permitiendo la ejercitación de metodologías de trabajo vinculadas a las formas profesionales del trabajo de un programador [5]. Por otro lado, y a raíz de la situación pandémica, se ha podido analizar cómo la estrategia del ABP resulta fácilmente adaptable a la virtualidad. En nuestro caso, esa adaptación ha implicado un rediseño particularmente de la metodología de evaluación y la exposición final de los proyectos, empleando una “Vidriera Virtual de Exposición de Proyectos” ver online (<https://sites.google.com/view/vidriera-proyectos-fceia/>), que nos permitió llevar adelante un trabajo de evaluación asíncrono entre pares y entre los docentes con excelentes resultados en su dinámica como propuesta de exposición y posterior repositorio de los trabajos [9]. Cabe mencionar por último que la presencia de plagios se ha reducido a cero durante la realización y concreción de los proyectos. Esta problemática es usual en los primeros años y es notable como con un abordaje creativo orientado a la programación grupal favoreció a la presentación de trabajos originales.

3 Trabajos Futuros

A partir de las experiencias realizadas, los aprendizajes alcanzados y las fortalezas detectadas [5], consideramos fundamental continuar con la implementación de esta estrategia en el redictado, ofreciendo a los y las estudiantes propuestas didácticas que les permitan resignificar lo trabajado en sus primeras cursadas. En esta dirección y con la misma filosofía de trabajo, se modificó el cursado del redictado de Programación II para LCC, donde se enseña a programar usando el lenguaje Python y C, para implementar en dicha cátedra también ABP sobre un trabajo integrador en el lenguaje Python [13]. Las experiencias ganadas como docentes sobre el uso y los alcances de la estrategia didáctica y el dominio de la técnica del ABP han permitido lograr nuevas propuestas pedagógicas con resultados igualmente de prometedores, que serán también documentados para su posterior análisis y difusión.

Referencias

1. Benotti, L., Echeveste, M., Schapachnik, F.: Despertando vocaciones en computación mediante el uso de autómatas de chat. En 6tas Jornadas de Vinculación Universidad Industria, 41JAIIO, 12-21 (2012).
2. Dapozo, G., Greiner, C., Pedrozo Petrazzini, G., Chiapello, J.: Vocaciones tic. ¿qué tienen en común los alumnos del nivel medio interesados por carreras de informática? En IX Congreso de Tecnología en Educación & Educación en Tecnología, 128-139 (2014).
3. Sánchez, P., Blanco, C.: Implantación de una metodología de aprendizaje basada en proyectos para una asignatura de Ingeniería del Software. En *Actas XVIII JENUI 2012*, Ciudad Real, Universidad Nacional de Cantabria, España (2012).
4. García Martín, J., Perez Martínez, J. : Aprendizaje basado en proyectos: método para el diseño de actividades. *Revista Tecnología, Ciencia y Educación*, 5, 37-63 (2018). Martínez Lopez, P.: Las Bases de la Programación. Publicado electrónicamente por la Universidad Virtual de Quilmes, La Plata, (2013).
5. Colussi, N., Viale, P., Monjelat, N.: Proyectos Grupales de Programación. Experiencias del ABP en el Aula Universitaria. En: Sgreccia, N. (ed.) *Jornadas de experiencias innovadoras en Educación en FCEIA*, 136-153. UNR, Rosario (2021)
6. Martínez Lopez, P.: Las Bases de la Programación. Publicado electrónicamente por la Universidad Virtual de Quilmes, La Plata, (2013).
7. Felleisen, M., Findler, R., Flatt, M., Krishnamurthi, S.: *How to Design Programs: An Introduction to Programming and Computing*. MIT Press, USA (2001).
8. Colussi, N., Viale, P.: Actividades de Programación Grupales para Primer año de la Licenciatura en Ciencias de la Computación - Experiencias Didácticas en el Aula. Versión Extendida. En *Jornadas de CyT de la UNR, (JorCyT)*, (2019).
9. Colussi, N., Viale, P., Monjelat, N.: Vidriera de proyectos: una modalidad de evaluación posible en tiempos de virtualidad. En *Actas de II WITE - TRANSFORMACIÓN DIGITAL. DESAFÍOS DE LA EDUCACIÓN SUPERIOR*, 1-10, (2021)
10. Wing, J.M.: Computational thinking. *Commun. ACM* , 49(3), 33-35, (2006).
11. Polya, G.: *How To Solve It: A New Aspect of Mathematical Method*. Princeton: University Press, (1973).
12. Thompson S.: Where do I begin? A problem solving approach in teaching functional programming. En Glaser, H., Hartel, P. y Kuchen, H. (eds.) *Programming Languages: Implementations, Logics, and Programs. PLILP 1997. Lecture Notes in Computer Science* (vol 1292). Springer, Berlin, Heidelberg, (1997).
13. Frydenberg, M., Mentzer, K.: From Engagement to Empowerment: Project-Based Learning in Python Coding Courses. *EDISG Conference, Information Systems & Computing Academic Professionals*, (2020).
14. Keogh S., Bradnum J., Anderson E.: Improving professionalism in first year computer science students: Teaching what can't be taught. En *Proceedings of the 3rd Conference on Computing Education Practice (CEP '19)*, 1-4, (2019).
15. Monjelat, N.; Colussi, N.; Viale, P.: Introducción a la Programación en carreras terciarias y universitarias de Computación: estado del arte en el contexto argentino. *Jornadas de Ciencia y Técnica de la UNR*, (2021).